

Kepler-sovellusprojekti

**Joonas Konki
Anu Koskela
Mikko Kuhno
Henrik Paananen
Atte Rätty**

Sovellusraportti

Julkinen
Versio 0.2.0
18.6.2015

**Jyväskylän yliopisto
Tietotekniikan laitos
Jyväskylä**

Hyväksyjä	Päivämäärä	Allekirjoitus	Nimenselvennys
Projektipäällikkö	__.__.2015		
Tilaaja	__.__.2015		
Ohjaaja	__.__.2015		

Tietoa dokumentista

Tekijät:

- Joonas Konki (JK) `joonas.konki@jyu.fi`
- Anu Koskela (AK) `anu.k.koskela@student.jyu.fi`
- Mikko Kuhno (MK) `mikko.kuhno@gmail.com`
- Henrik Paananen (HP) `henrik.j.paananen@student.jyu.fi`
- Atte Rätty (AR) `atte.t.raty@student.jyu.fi`

Dokumentin nimi: Kepler-projekti, Sovellusraportti

Sivumäärä: 59

Tiivistelmä: Kepler-projekti kehitti keväällä 2015 Jyväskylän yliopiston fysiikan laitokselle WWW-sovelluksen oppilaslaboratorion töiden mittausvuorojen varaamiseen. Sovellusraportissa kuvataan projektissa kehitetyn tietojärjestelmän käyttöliittymää, sovelluksen kokonaisrakennetta, tavoitteiden toteutumista, tietokannan rakennetta sekä puutteellisia ja heikkoja toteutusratkaisuja.

Avainsanat: Jatkokehitys, kokonaisrakenne, komponentit, käyttöliittymä, käytänteet, OAuth, palvelin, Pyramid, rajapinnat, REST, tavoitteet, tietojärjestelmä, tietokanta, ylläpito.

Muutoshistoria

Versio	Päivämäärä	Muutokset	Tekijät
0.0.1	10.6.2015	Dokumentin laatiminen aloitettiin.	AR
0.0.2	13.6.2015	Käyttöliittymä näkymien kuvailua.	AR
0.1.0	14.6.2015	Dokumentin puuttuvien lukujen ja alalukujen tekoa.	AR
0.1.1	15.6.2015	Dokumentin puuttuvien osien täydentämistä ja kieliäsun parantelua.	AR
0.1.2	16.6.2015	Dokumentin puuttuvien osien täydentämistä, kieliäsun parantelua ja ohjaajan palautteen mukaisten korjausten tekemistä.	AR
0.2.0	17.6.2015	Ohjaajan esittämien korjausten tekoa.	AR
0.2.1	18.6.2015	?????????	AR
Versio	Päivämäärä	Muutokset	Tekijät

Tietoa projektista

Kepler-projekti kehitti keväällä 2015 Jyväskylän yliopiston fysiikan laitokselle WWW-sovelluksen oppilaslaboratorion töiden mittausvuorojen varaamiseen.

Tekijät:

- Joonas Konki (JK) `joonas.konki@jyu.fi`
- Anu Koskela (AK) `anu.k.koskela@student.jyu.fi`
- Mikko Kuhno (MK) `mikko.kuhno@gmail.com`
- Henrik Paananen (HP) `henrik.j.paananen@student.jyu.fi`
- Atte Rätty (AR) `atte.t.raty@student.jyu.fi`

Tilaaaja:

- Sakari Juutinen `sakari.juutinen@phys.jyu.fi`
- Panu Rahkila `panu.rahkila@jyu.fi`

Ohjaajat:

- Petri Partanen `petri.m.partanen@student.jyu.fi`
- Jukka-Pekka Santanen `santanen@mit.jyu.fi`

Yhteystiedot:

- Sähköpostilistat: `keplerit@korppi.jyu.fi` ja `keplerit_opetus@korppi.jyu.fi`
- Sähköpostiarkistot: `http://korppi.jyu.fi/kotka/servlet/list-archive/keplerit/` ja `http://korppi.jyu.fi/kotka/servlet/list-archive/kepler_opetus/`

Sisältö

1	Johdanto	1
2	Termit	2
2.1	Aihealueen termejä	2
2.2	Ohjelmistoja ja teknisiä termejä	3
3	Käyttöliittymä	5
3.1	Sovelluksen näkymät	5
3.2	Ylänavigointialue	7
3.2.1	Järjestelmän nimi ja kuvaus	7
3.2.2	Kielen valinta	7
3.2.3	Apua-näkymä	7
3.2.4	Käyttäjä-valikko	8
3.2.5	Omat tiedot	9
3.2.6	Kirjaudu ulos	10
3.3	Navigointipalkki	10
3.4	Auki oleva näkymä	10
3.5	Opiskelijan näkymät	10
3.5.1	Ilmoitustaulu	10
3.5.2	Uusi varaus	11
3.5.3	Omat varaukset	14
3.5.4	Omat ryhmät	16
3.6	Ohjaajan näkymät	17
3.6.1	Ohjausvuorot	18
3.7	Pääkäyttäjän näkymät	19
3.7.1	Hallitse vuoroja	19
3.7.2	Kurssit	21
3.7.3	Työt	23
3.7.4	Resurssit	26
3.7.5	Käyttäjäroolit	28
3.7.6	Muokkaa ilmoitustaulua	28
3.8	Kirjautumissivu	30
4	Sovelluksen kokonaisrakenne ja rajapinnat	34
4.1	Sovelluksen kokonaisrakenne	34

Kepler-projekti	Sovellusraportti 0.2.0	Julkinen
4.2	Ulkopuoliset komponentit	36
4.3	Olennaisimmat muutokset toteutusratkaisuihin	37
4.4	Tietokannan rakenne	38
4.5	Muutettavat asetukset	41
4.6	Ulkoiset rajapinnat	45
4.6.1	Korppi-kalenteri	45
4.6.2	OAuth	46
5	Tavoitteiden toteutuminen	47
5.1	Vaatimusten toteutuminen	47
5.2	Suoritettavat testaukset ja niiden tulokset	48
5.3	Heikot ja puutteelliset toteutusratkaisut	49
5.4	Sovelluksen yleiset jatkokehitysideat	51
5.5	Opiskelijan näkymiin liittyvät jatkokehitysideat	52
5.6	Ohjaajan näkymiin liittyvät jatkokehitysideat	52
5.7	Pääkäyttäjän näkymiin liittyvät jatkokehitysideat	53
5.8	Tietojärjestelmän yleiskäyttöisyys	54
6	Ohjeita ylläpitäjälle ja jatkokehittäjälle	55
6.1	Ohjeita jatkokehittäjälle	55
6.2	Ohjeita ylläpitäjälle	57
7	Yhteenveto	58
	Lähteet	59

1 Johdanto

Jyväskylän yliopiston fysiikan laitoksen oppilaslaboratoriossa opiskelijat voivat suorittaa kokeelliset mittaukset fysiikan perus- ja aineopintotasoihin kursseihin liittyviin laboratoriotöihin. Fysiikan laboratoriotyöt ovat tärkeä osa fysiikan kurssien sisältöä, missä luennoilla opetettua teoriaa pääsee soveltamaan käytäntöön. Kepler-projekti kehitti WWW-sovelluksen fysiikan laitoksen oppilaslaboratorion laboratoriotöiden varausten hallintaan. Varausjärjestelmää tulevat käyttämään pääasiassa fysiikan perus- ja aineopintokurssien opiskelijat, laboratoriovuoroja ohjaavat assistentit sekä varausjärjestelmän pääkäyttäjät. Kehitettävä sovellus korvaa nykyisen WWW-selaimella käytettävän Brahe2007-varausjärjestelmän.

Sovellusprojektissa kehitetyn sovelluksen käyttöliittymän ulkoasu päivitettiin nykyaikaisemmaksi. Vanhasta varausjärjestelmästä kokonaan puuttuva pääkäyttäjän näkymä ja osa sen toiminnoista toteutettiin uudessa järjestelmässä. Lisäksi projektin alussa kartoitettiin muiden yliopiston laitosten varausjärjestelmätarpeet, jotka huomioitiin uuden järjestelmän kehittämisessä ja suunnittelussa.

Sovellusraportin laatimisessa on hyödynnetty Liikkuva-projektin sovellusraporttia [1] sekä Kepler-projektin projektiraporttia [5], projektisuunnitelmaa [6] ja vaatimusmäärittelyä [7]. Muita Kepler-projektin dokumentteja ovat käyttöliittymän [?] ja palvelimen luokkadokumentaatio [?], järjestelmätestausraportti [?], käytettävyyssuunnitelma [?] ja käytettävyydestä tehty muistio [?].

Sovellusraportti muodostuu seitsemästä luvusta. Luvussa 2 kuvataan olennaisia termejä. Luvussa 3 kuvataan sovelluksen käyttöliittymää näkymittäin. Luvussa 4 kuvataan sovelluksen kokonaisrakennetta ja rajapintoja. Luvussa 5 kuvataan heikkoja ja puutteellisia toteutusratkaisuja sekä jatkokehitysideoita. Luvussa 6 annetaan ohjeita ylläpitäjälle ja jatkokehittäjälle.

2 Termit

Luvussa kuvataan projektissa käytettäviä aihealueen, tietojärjestelmän ja toteutus-tekniikoiden termejä.

2.1 Aihealueen termejä

Projektin aihealueen termejä ovat seuraavat:

Assistentti	on fysiikan laboratoriotöiden ohjaajasta käytetty nimitys. Hän on perus-, aine- tai syventäviin opintoihin liittyvien kurssien laboratoriotöitä laboratoriovuoron aikana ohjaava henkilö.
Laboratoriotyö	on mittausvuorolla suoritettava johonkin kurssiin liittyvä työ.
Laboratoriovuoro	on yleensä neljän tunnin mittainen assistentin ohjaama tilaisuus, jossa laboratoriotöihin liittyvät mittaukset voidaan suorittaa.
Mittauslaitteisto	on yhdestä tai useammasta mittalaitteesta koostuva laitteisto, jolla laboratoriotyön mittaukset suoritetaan.
Mittausryhmä	on opiskelijajoukko, joka on varannut ajan laboratoriotyön mittaamiseen jollekin ohjatulle laboratoriovuorolle.
Mittausvuoro	on opiskelijan joltakin laboratoriovuorolta varaama aika jonkin tietyn laboratoriotyön mittaamiseen.
Ohjaaja	on assistentin käyttäjäroolin nimi kehitetyssä varausjärjestelmässä.
Ohjausvuoro	on ohjaajan laboratoriovuoro, jolle hänet on asetettu ohjaajaksi.
Opiskelija	on mittausvuoroja varaavan opiskelijan käyttäjäroolin nimi kehitetyssä varausjärjestelmässä.

Oppilaslaboratorio	on fysiikan laitoksen oppilaslaboratorio, jonka vuorojen varaukseen järjestelmä kehitettiin.
Pääkäyttäjä	on oppilaslaboratorion johtajan käyttäjäroolin nimi kehitetyssä varausjärjestelmässä. Pääkäyttäjä on henkilö, joka hallitsee toisten käyttäjien rooleja ja lisää laboratoriotöitä järjestelmään.

2.2 Ohjelmistoja ja teknisiä termejä

Dokumentissa käytettyjä teknisiä termejä ovat seuraavat:

Back end	on ohjelmiston se osa, joka on kauimpana ohjelmiston käyttäjästä. WWW-sovelluksessa se on palvelinpuolen ohjelmisto.
Bootstrap	on WWW-dokumenttien ulkoasun muokkausta helpottava, HTML-kuvauskieltä ja CSS-tyylikieltä käyttävä kirjasto.
CSRF	on tietoturvahyökkäys, joka pakottaa käyttäjän suorittamaan hänen haluamattaan toimintoja WWW-sovelluksessa, johon hän on kirjautunut. Termi tulee sanoista <i>cross-site request forgery</i> .
CSS	on tyylikieli WWW-dokumenttien ulkoasun määrittämiseen.
Front end	on ohjelmiston se osa, joka on lähimpänä ohjelmiston käyttäjää. Käytännössä front end sisältää sovelluksen käyttöliittymän ja sen toimintalogiikan.
HTML5	on uusin versio WWW-dokumenttien HTML-kuvauskielestä.
HTTPS	on WWW:ssä suojattuun tietoliikenteeseen käytetty protokolla.
iCalendar	on kalenteritiedostomuoto, jolla kalenterisovellukset pystyvät vaihtamaan kalenteritietoja keskenään.

JavaScript	on WWW-sovelluksissa käytetty ja WWW-selaimissa toimiva dynaamisesti tyypitetty ohjelmointikieli.
jQuery	on WWW-dokumenttien dynaamisen sisällön käsittelyä helpottava avoimen lähdekoodin lisenssin alainen JavaScript-kirjasto.
JYU-tunnukset	ovat Jyväskylän yliopiston IT-palveluiden käyttäjän henkilökohtaiset käyttäjätunnukset.
Korppi	on Jyväskylän yliopiston opintotietojärjestelmä.
Käyttöliittymä	on ohjelmiston osa, jonka kautta käyttäjä käyttää ohjelmistoa.
OAuth2	on sovelluksessa autentikointiin käytettävä protokolla.
ORM	on tekniikka, jolla tietokannan tietoja pystytään helposti käsittelemään suoraan ohjelmointikielen olioiden kautta.
Pyramid	on Python-ohjelmointikielillä toteutettu minimalistinen WWW-sovelluskehys.
Python	on dynaamisesti tyypitetty korkeantason ohjelmointikieli.
Sovelluskehys	on ohjelman rungon muodostava teknologia.
SQL	on yleisesti käytetty relaatiotietokantastandardi.
SQLAlchemy	on avoimen lähdekoodin SQL- ja ORM-kirjasto Python-kielille.
Tietokanta	on tietovarasto, joka tyypillisesti mallintaa jollain tapaa tiedon yhteyksiä toisiin tietoihin. Tällöin puhutaan relaatiotietokannasta.
WWW-sovellus	on WWW-tekniikoita hyödyntävä ja selaimella käytettävä sovellus.
WWW-sovelluskehys	on sovelluskehys, joka on suunniteltu nopeuttamaan WWW-sovellusten kehittämistä.

3 Käyttöliittymä

Luvussa esitellään sovelluksen käyttöliittymän eri näkymät sekä niihin liittyvät toiminnot ja niiden väliset suhteet.

3.1 Sovelluksen näkymät

Kuvassa 3.1 havainnollistetaan sovelluksen yleistä näkymää, kun on kirjautuneena sisään pääkäyttäjänä.



Copyright © 2015 Kepler project group members
Anna palautetta Apua

Kuva 3.1: Keplerin yleisnäkyminen pääkäyttäjänä

Sovelluksen käyttöliittymä sisältää koostuu seuraavista osista:

- ylänavigointialue (osoitettu kirjaimella A),
- järjestelmän nimi ja kuvaus (A1),

- kielen valinta (A2),
- Apua-näkymä (A3),
- käyttäjä-valikko (A4),
- navigointipalkki (B),
- valitun näkymän korostus (B1),
- opiskelijan näkymiä (B2, sisältää myös tällä hetkellä korostetun Ilmoitustaulu-näkymän),
- ohjaajan näkymän (B3),
- pääkäyttäjän näkymiä (B4),
- auki oleva näkymä (C)

3.2 Ylänavigointialue

Kepler-järjestelmän kaikissa näkymissä (kirjautumissivua lukuun ottamatta) on nähtävissä ylänavigointialue.

Kuva 3.2: Yläpalkki

Ylänavigointialue koostuu neljästä osasta, jotka ovat vasemmalta oikealle lueteltuna: järjestelmän nimi ja kuvaus, kielen valinta, ohjeisiin vievä painike ja käyttäjävalikko.

3.2.1 Järjestelmän nimi ja kuvaus

Järjestelmän nimi ja kuvaus kertovat käyttäjälle hänen käyttämänsä järjestelmän nimen ja lyhyen kuvauksen sen käyttötarkoituksesta.

3.2.2 Kielen valinta

Kielen valinnassa esitetään kielivaihtoehdot lippuina. Lippua painamalla järjestelmän käyttämä kieli vaihtuu tämän istunnon ajaksi. Pysyvän kielivalinnan voi tehdä Omat tiedot-näkymässä.

3.2.3 Apua-näkymä

Apua-näkymään pääsee painamalla kielen valinnan vieressä esitettyä kysymysmerkkipainiketta. Apua-näkymä on koko järjestelmälle yleinen (se ei siis muutu sivukohtaisesti).

- Uusi varaus
- Omat varaukset
- Omat ryhmät
- Ohjausvuorot
- Hallitse vuoroja
- Kurssit
- Työt
- Resurssit
- Käyttäjäroolit
- Muokkaa ilmoitustaulua

Kepler apua

Kepler on ensisijaisesti opiskelijoille suunniteltu laboratorioiden varausjärjestelmä. Ohjelmiston päätarkoitus on antaa käyttäjälle looginen tapa muodostaa ryhmiä ja varata laboratoriotyöaikoja.

Opiskelijalla on käytössä neljä eri näkymää:

- Ilmoitustaulu
- Uusi varaus
- Omat varaukset
- Omat ryhmät

Ilmoitustaulu:

Ilmoitustaululta näet tärkeimmät uutiset liittyen varauksiin.

Uusi varaus:

Tässä näkymässä voidaan varata haluttu laboratoriovuoro. Ensimmäiseksi listasta etsitään ja valitaan haluttu työ, tämän jälkeen listan alapuolelle ilmestyy tärkeimmät tiedot työstä, sekä valintalaatikko mille ryhmälle vuoron haluaa varata. Alimpana näkyy oletusarvoisesti kalenterinäkymä, missä varattavat vuorot näkyvät **talla** värillä. Vuoro valitaan klikkaamalla haluttua vapaata aikaa ja tämän jälkeen painamalla "tee varaus" painiketta.

Vaihtoehtoisesti vuoron voi varata listanäkymässä klikkaamalla haluttua aikaa ja painamalla "tee varaus" painiketta.

Lisätietokenttään suositellaan kirjoitettavaksi mahdolliset rajoitteet laboratoriotyön tekemisessä kuten kieli tai liikuntarajoitteet.

Omat varaukset:

Uusi varaus kohdassa tehdyt varaukset näkyvät täällä oletusarvoisesti listanäkymänä. Listanäkymän osaa klikkaamalla voi perua vuoron, mikäli vuoron alkuun on aikaa yli 48 tuntia. Kalenterinäkymässä varatut vuorot näkyvät **talla** värillä. Varaus voidaan perua kalenterinäkymässä valitsemalla ensin peruttava vuoro ja tämän jälkeen perumalla se "peru valittu varaus" painikkeella.

Omat ryhmät:

Omissa ryhmissä voit katsella, poistaa ja luoda uusia ryhmiä. Ryhmän voi poistaa vain jos sillä ei ole keskeneräisiä laboratoriovarauksia, ja jos on luonut ryhmän itse. Uuden ryhmän luomisessa käyttäjän haussa tulee kirjoittaa sähköpostiosoite tai käyttäjätunnus **kokonaisuudessaan**. Huomioi isot ja pienet kirjaimet.

Kalenterin selitteet:

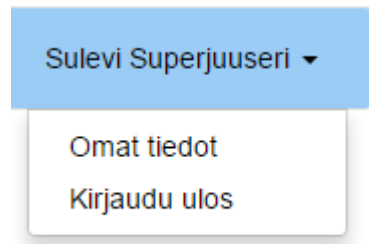
- Varattavissa oleva aika
- Omat ohjausvuorot
- Varattu vuoro
- Valittu vuoro
- Valittu poistettava vuoro / peruttu vuoro

Kiitos, kun jaksoit lukea ohjeet. Tässä pieni säästöpossu palkinnoksi: 🐾

Kuva 3.3: Apua -näkyvä

3.2.4 Käyttäjä-valikko

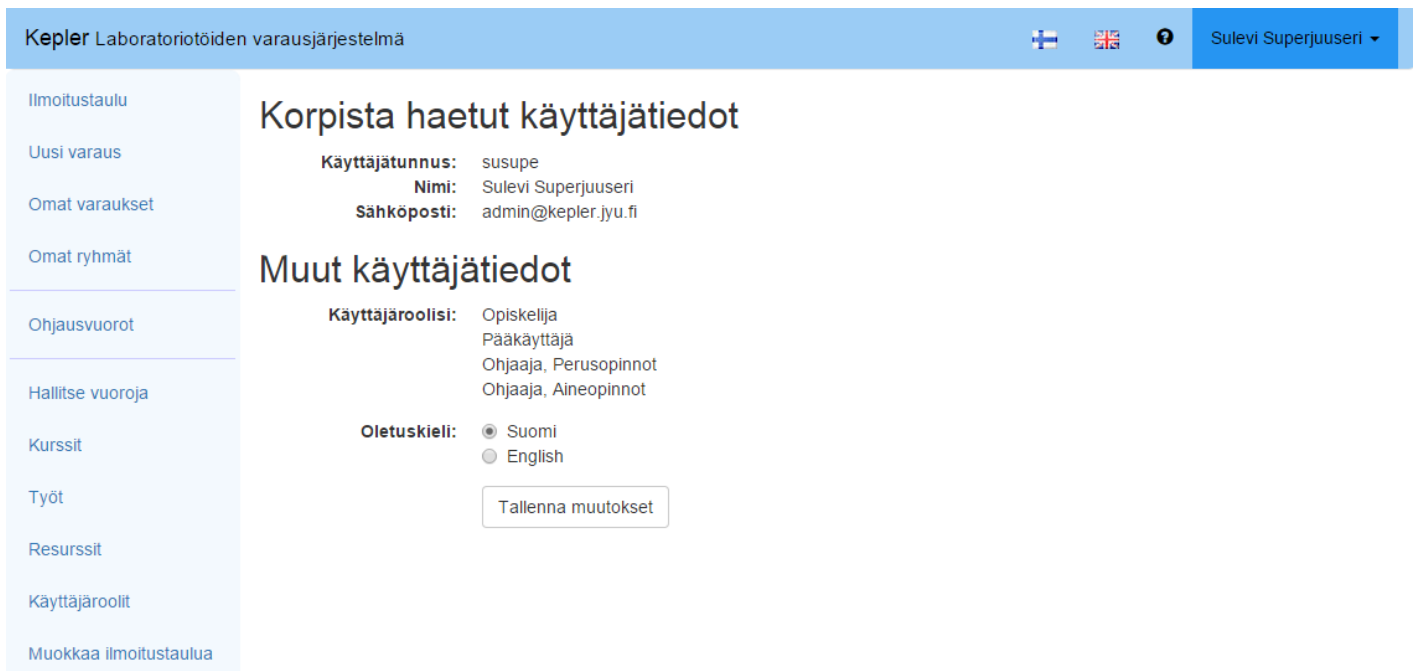
Käyttäjä voi omaa nimeään painamalla avata valikon, josta paljastuu erilaisia toimenpiteitä.



Kuva 3.4: Käyttäjä-valikko

3.2.5 Omat tiedot

Omat tiedot -painiketta painamalla aukeaa seuraavanlainen näkymä.



Kuva 3.5: Omat tiedot

Tässä näkymässä käyttäjä näkee ne tietonsa, jotka tietojärjestelmästä löytyvät. Käyttäjä voi vaihtaa oletuskieltään valitsemalla haluamansa kielen ja painamalla Tallenna muutokset-painiketta.

3.2.6 Kirjaudu ulos

Kirjaudu ulos-painiketta painamalla käyttäjä voi kirjautua ulos tietojärjestelmästä.

3.3 Navigointipalkki

Navigointipalkkia (kirjain B kuvassa 3.1) käytetään eri näkymien välillä siirtymiseen. Sen sisältö muuttuu sisäänkirjautuneen käyttäjän roolien mukaan. Kuvassa 3.1 sisäänkirjautuneella käyttäjällä on opiskelijan, ohjaajan ja pääkäyttäjän roolit, joten hän näkee navigointipalkin kokonaisuudessaan.

Navigointipalkissa (ja ylänavigointipalkissa) on auki oleva näkymä korostettuna tummemmalla taustavärillä.

3.4 Auki oleva näkymä

Auki olevassa näkymässä (kirjain C kuvassa 3.1) esitetään kulloinkin auki olevan näkymän toiminnallisuus. Suurin osa tietojärjestelmän toiminnallisuuksista suoritetaan käyttäen kulloinkin valittuna olevaa näkymää.

3.5 Opiskelijan näkymät

Opiskelijan roolin omaavalla käyttäjällä on oletuksena käytössään seuraavat näkymät: Ilmoitustaulu, Uusi varaus, Omat varaukset ja Omat ryhmät.

3.5.1 Ilmoitustaulu

Ilmoitustaululla käyttäjät näkevät pääkäyttäjän järjestelmään lisäämiä ilmoituksia. Ominaisuus havaittiin hyväksi Kepleriä edeltäneessä Brahe2007 -järjestelmässä.

The screenshot shows the Kepler Laboratory Reservation System interface. At the top, there is a blue header with the text 'Kepler Laboratoriotöiden varausjärjestelmä' on the left and flags for Finnish and English, along with a user profile 'Sulevi Superjuuseri' on the right. Below the header is a sidebar menu with the following items: 'Ilmoitustaulu' (highlighted), 'Uusi varaus', 'Omat varaukset', 'Omat ryhmät', 'Ohjausvuorot', 'Hallitse vuoroja', 'Kurssit', 'Työt', 'Resurssit', 'Käyttäjäroolit', and 'Muokkaa ilmoitustaulua'. The main content area is titled 'Ilmoitustaulu' and contains two notices. The first notice is titled 'Lisää vuoroja tulossa!' and is dated '13.6.2015 12.16'. The text of the notice states: 'Laitoksella on ollut suunnitelmassa uusien vuorojen lisääminen laboratoriovuorojen kiertoon. Nyt asia on päätetty. Opiskelijoilta otetaan vastaan toiveita vuorojen ajankohdista.' The second notice is titled 'Muistakaa laskimenne!' and is dated '13.6.2015 12.14'. The text of the notice states: 'Muistakaa ottaa labroihiin mukaan laskimenne! Labroissa varalaskimia on tarjolla hyvin rajoitetusti.'

Kuva 3.6: Ilmoitustaulu

3.5.2 Uusi varaus

Uusi varaus-näkymässä käyttäjät voivat tehdä uusia varauksia itselleen ja ryhmilleen. Varauksen tehdäkseen käyttäjän pitää valita, mitä työtä varten ja kenelle varaus tehdään. Varauksen ajankohta voidaan valita käyttäen joko kalenteri- tai listanäkymää.

Kepler Laboratoriotöiden varausjärjestelmä 🇫🇮 🇬🇧 👤 Sulevi Superjuuseri ▾

Uusi varaus

Etsi...

Valitse työ:

- FYSP101/1, Nopeuden mittaus
- FYSP101/K1, Kinematiikan kuvaajat
- FYSP101/K2, Heittoliike
- FYSP101/K3, Törmäykset ilmaradalla
- FYSP102/1, Vieriminen**
- FYSP102/2, Kiertoheiluri
- FYSP102/2a, Kiertoheiluri (ajanotto kellolla)
- FYSP102/K1, Pyörimisliike ja keskihakuvoima
- FYSP102/K2, Kierroheijuri

Työn nimi
FYSP102/1, Vieriminen

Kuvaus
Työn vastuuhenkilöt ovat A. Assari ja O. Ohjaaja. Työn ohjeet: www

Mittausryhmän maksimikoko
2 jäsentä

Kenelle varaus tehdään:
Sulevi Superjuuseri ▾

Ilmoitustaulu

- Uusi varaus**
- Omat varaukset
- Omat ryhmät
- Ohjausvuorot
- Hallitse vuoroja
- Kurssit
- Työt
- Resurssit
- Käyttäjäroolit
- Muokkaa ilmoitustaulua

Kuva 3.7: Uuteen varaukseen liittyvien valintaperusteiden valinta

Mittausryhmän maksimikoko

2 jäsentä

Kenelle varaus tehdään:

Sulevi Superjuuseri

Valitse sopiva mittausvuoro:

Vaihda listanäkymään

Päivä	Viikko	Kuukausi	15 — 21.6.2015		Tänään	<	>
	ma - 15.06	ti - 16.06	ke - 17.06	to - 18.06	pe - 19.06		
8:00							
9:00							
10:00							
11:00							
12:00		12.00 - 16.00 Vieriminen					
13:00							
14:00							
15:00							
16:00				16.00 - 20.00 Vieriminen			
17:00							
18:00							
19:00							

Anna halutessasi lisätietoja:

Tee varaus

Kuva 3.8: Uuden varauksen ajankohdan valinta kalenterin avulla

Mittausryhmän maksimikoko

2 jäsentä

Kenelle varaus tehdään:

Sulevi Superjuuseri ▼

Valitse sopiva mittausvuoro:

Vaihda kalenterinäkymään

Ohjausvuorot: 16.6.2015 12.00 - 16.00, Perusopinnot 18.6.2015 16.00 - 20.00, Perusopinnot**Anna halutessasi lisätietoja:**

Tee varaus

Kuva 3.9: Uuden varauksen ajankohdan valinta listan avulla

3.5.3 Omat varaukset

Tässä näkymässä käyttäjä näkee omat ja ryhmiensä varaukset, niihin liittyviä tietoja sekä pystyy perumaan niitä (mikäli kyseinen vuoro on vielä peruttavissa). Käyttäjä voi myös valita tarkasteltavan ajanjakson.

Kepler Laboratoriotöiden varausjärjestelmä 🇫🇮 🇬🇧 🔔 Sulevi Superjuuseri ▾

Ilmoitustaulu
Uusi varaus
Omat varaukset
Omat ryhmät
Ohjausvuorot
Hallitse vuoroja
Kurssit
Työt
Resurssit
Käyttäjäroolit
Muokkaa ilmoitustaulua

Omat varaukset

Menneet vuorot Menossa nyt Tulevat vuorot Vaihda kalenterinäkymään

Näytetään vuorot ajalta: ▾

>	20.5.2015 16.00 - 20.00	FYSP101/1, Nopeuden mittaus
▼	22.5.2015 12.00 - 16.00	FYSP102/1, Vieriminen
Ryhmä: Sulevi Superjuuseri		
Peru varaus		
>	25.5.2015 8.00 - 12.00	FYSP102/1, Vieriminen Tämä vuoro on peruttu!
>	25.5.2015 8.00 - 12.00	FYSP101/K1, Kinematiikan kuvaajat Tämä vuoro on peruttu!
>	27.5.2015 16.00 - 20.00	FYSP101/K3, Törmäykset ilmaradalla Tämä vuoro on peruttu!
>	1.6.2015 8.00 - 12.00	FYSP102/1, Vieriminen
>	3.6.2015 16.00 - 20.00	FYSP101/K2, Heittoliike
>	8.6.2015 8.00 - 12.00	FYSP103/K3, Braggin diffraktio
>	16.6.2015 12.00 - 16.00	FYSP102/1, Vieriminen

Copyright © 2015 Kepler project group members
Anna palautetta Apua

Kuva 3.10: Omat varaukset listamuodossa

Omat varauksensa on mahdollista nähdä myös kalenterimuodossa.

Kepler Laboratoriotöiden varausjärjestelmä 🇫🇮 🇬🇧 👤 Sulevi Superjuuseri ▾

Ilmoitustaulu
Uusi varaus
Omat varaukset
Omat ryhmät

Ohjausvuorot
Hallitse vuoroja
Kurssit
Työt
Resurssit
Käyttäjäroolit
Muokkaa ilmoitustaulua

Omat varaukset

Vaihda listanäkymään

Päivä Viikko Kuukausi

8 — 14.6.2015 Tänään < >

	ma - 08.06	ti - 09.06	ke - 10.06	to - 11.06	pe - 12.06
8:00	08.00 - 12.00 FYSP103/K3 Braggin diffraktio Henkilökohtainen varaus				
9:00					
10:00					
11:00					
12:00					
13:00					
14:00					
15:00					
16:00					
17:00					
18:00					
19:00					

Peru valittu varaus

Kuva 3.11: Omat varaukset kalenterimuodossa

3.5.4 Omat ryhmät

Omat ryhmät-näkymässä käyttäjä voi tarkastella ryhmiä, joihin hän kuuluu. Hän voi poistaa luomiaan ryhmiä, joiden omistaja hän on, ja hän voi luoda uusia ryhmiä.

Kepler Laboratoriotöiden varausjärjestelmä

Ilmoitustaulu
Uusi varaus
Omat varaukset
Omat ryhmät

Omat ryhmät

▼ Keplot 2

Olli Opiskelija	Ryhmän omistaja
Kalle Kalastaja	

Poista ryhmä

▼ testiryhma 2

Tee uusi ryhmä

Kuva 3.12: Omat ryhmät

Uuden ryhmän luominen

Ryhmän nimi

Hae käyttäjä

Ryhmän jäsenet

- Olli Opiskelija

Kuva 3.13: Uuden ryhmän luominen

3.6 Ohjaajan näkymät

Ohjaajan roolin omaavalla käyttäjällä on oletuksena käytössään (vain) Ohjausvuorot-näkymä. Huomaa, että ainakin Jyväskylän yliopiston fysiikan laitoksen käyttötapauksessa oikeastaan kukaan ei kuitenkaan ole rooliltaan pelkästään ohjaaja, sillä ohjaajat ovat joko laitoksen opiskelijoita tai henkilökuntaa.

3.6.1 Ohjausvuorot

Ohjausvuorot-näkymässä käyttäjä näkee omat ohjausvuoronsa. Käyttäjä voi valita haluaako hän nähdä menneitä, meneillään olevia tai tulevia vuoroja. Käyttäjä voi myös nähdä listauksen ohjausvuorojensa osallistujista ja näiden osallistujien yhteystiedot (eli sähköpostiosoitteen). Käyttäjä voi nähdä ohjausvuoronsa lista- tai kalenterinäkymänä ja hän voi valita, haluaako nähdä myös muiden ohjaajien ohjausvuorot.

Kepler Laboratoriotöiden varausjärjestelmä

Ohjausvuorot

Menneet vuorot Menossa nyt Tulevat vuorot

Näytä omat vuorot Näytä kaikkien vuorot

Vaihda kalenterinäkymään

Näytetään vuorot ajalta: Kuukausi

16.6.2015 12.00 - 16.00	Perusopinnot, Maria Assari	2	
FYSP102/1, Vieriminen			
<input checked="" type="checkbox"/> Sulevi Superjuuseri			
FYSP102/1, Vieriminen			
<input checked="" type="checkbox"/> Olli Opiskelija			
Yhteystiedot	00 - 20.00	Perusopinnot, Maria Assari	0
Ei yhteystietoja	0 - 12.00	Perusopinnot, Maria Assari	0
19.6.2015 12.00 - 16.00	Perusopinnot, Maria Assari	0	

Kuva 3.14: Ohjausvuorot listana

Kepler Laboratoriotöiden varausjärjestelmä 🇫🇮 🇬🇧 🔔 Maria Assari ▾

Ilmoitustaulu
Uusi varaus
Omat varaukset
Omat ryhmät
Ohjausvuorot

Ohjausvuorot

Menneet vuorot Menossa nyt Tulevat vuorot

Näytä omat vuorot Näytä kaikkien vuorot

Vaihda iistanäkymään

Näytetään vuorot ajalta:

15 — 21.6.2015

	ma - 15.06	ti - 16.06	ke - 17.06	to - 18.06	pe - 19.06
8:00					
9:00					
10:00					
11:00					
12:00		12.00 - 16.00 Ohjausvuoro			
13:00		Ohjaajat: Anne Maria			
14:00					
15:00					
16:00					
17:00					
18:00					
19:00					

Kuva 3.15: Ohjausvuorot kalenterissa

3.7 Pääkäyttäjän näkymät

Pääkäyttäjän roolin omaavalla käyttäjällä on oletuksena käytössään seuraavat näkymät: Hallitse vuoroja, Kurssit, Työt, Resurssit, Käyttäjäroolit ja Muokkaa ilmoitustaulua.

3.7.1 Hallitse vuoroja

Hallitse vuoroja-näkymässä käyttäjä näkee kaikki Kepler-järjestelmän vuorot. Hän voi rajata tarkasteltavien vuorojen ajankohdaksi menneet, menossa olevat tai tulevat vuorot. Vuorosta esitetään tietoina mihin opintokokonaisuuteen vuorolla tehtävät laboratoriotyöt kuuluvat, ohjaajat ja lista osallistujista yhteystietoineen. Vuoroja

pystyy myös perumaan käyttämällä Peru-painiketta.

Tarkoituksena oli myös toteuttaa mahdollisuus esittää vuorot kalenterimuodossa ja tarjota mahdollisuus vuorojen muokkaamiseen, mutta nämä ominaisuudet sovittiin tilaajan kanssa jätettävän jatkokehitykseen.

The screenshot shows the 'Kepler Laboratoriotöiden varausjärjestelmä' (Kepler Laboratory Work Booking System) interface. The main heading is 'Hallitse ohjausvuoroja' (Manage shift assignments). There are filters for 'Menneet vuorot' (Past shifts), 'Menossa nyt' (Currently ongoing), and 'Tulevat vuorot' (Upcoming shifts), all of which are checked. A button 'Vaihda kalenterinäkymään' (Switch to calendar view) is visible. Below the filters, there is a dropdown menu for 'Näytetään vuorot ajalta:' (Show shifts from:) set to 'Kuukausi' (Monthly). A 'Lisää uusi vuoro...' (Add new shift...) button is present. The main content is a list of shifts with columns for date, time, location, and a count. The last shift is marked as cancelled.

Time	Location	Count
11.5.2015 8.00 - 12.00	Perusopinnot, Kalle Kalastaja	2
11.5.2015 8.00 - 12.00	Aineopinnot, Kalle Kalastaja	1
FYSA242/K1, Terminen elektroniemissio Olli Opiskelija		
12.5.2015 12.00 - 16.00	Aineopinnot, Kalle Kalastaja	1
13.5.2015 16.00 - 20.00	Perusopinnot, Kalle Kalastaja	0
15.5.2015 12.00 - 16.00	Aineopinnot, Kalle Kalastaja	1
18.5.2015 8.00 - 12.00	Perusopinnot, Kalle Kalastaja	2
18.5.2015 8.00 - 12.00	Aineopinnot, Kalle Kalastaja	1
19.5.2015 12.00 - 16.00	Aineopinnot, Kalle Kalastaja	1
20.5.2015 16.00 - 20.00	Perusopinnot, Kalle Kalastaja	1
22.5.2015 12.00 - 16.00	Aineopinnot, Kalle Kalastaja	1
25.5.2015 8.00 - 12.00	Perusopinnot, Kalle Kalastaja	0

Kuva 3.16: Hallitse vuoroja

16.6.2015 12.00 - 16.00 Perusopinnot, Maria Assari 2	
FYSP102/1, Vieriminen	<input type="checkbox"/> Sulevi Superjuuseri
FYSP102/1, Vieriminen	<input type="checkbox"/> Olli Opiskelija
Muokkaa Poista	

Kuva 3.17: Yksittäisen vuoron toiminnot

3.7.2 Kurssit

Kurssit-näkymässä listataan tietojärjestelmän sisältämät kurssit. Listattujen kurs-
sien tietoja voi myös muokata oikean reunan syöttökentillä ja painamalla Tallenna
muutokset-painiketta. Vanhoja kursseja pystyy poistamaan ja uusia kursseja luo-
maan.

Kursseihin liittyvien (laboratorio)töiden tietoja voi muokata Kurssit-näkymässä ja
kurssiin voi lisätä siihen liittyviä töitä.

Kepler Laboratoriotöiden varausjärjestelmä 🇫🇮 🇬🇧 👤 Sulevi Superjuuseri ▾

Hallitse kursseja

Etsi...

- FYSA210
- FYSA220
- FYSA230
- FYSA241
- FYSA242
- FYSP101
- FYSP102
- FYSP103
- FYSP104
- FYSP105
- FYSP106
- FYSP107
- FYSP110

Kurssin koodi: FYSA210

Opintokokonaisuus: Aineopinnot ▾

Työt:

K1	Hitausmomentti
1	Kytetty värähtelijä
K2	Kääntöheiluri
2	Pyörivä koordinaatisto

[+ Lisää työ](#)

[Tallenna muutokset](#) [Poista kurssi](#)

[+ Lisää uusi kurssi](#)

Kuva 3.18: Hallitse kursseja

Kepler Laboratoriotöiden varausjärjestelmä 🇫🇮 🇬🇧 👤 Sulevi Superjuuseri ▾

Hallitse kursseja

Etsi...

- FYSA210
- FYSA220
- FYSA230
- FYSA241
- FYSA242
- FYSP101
- FYSP102
- FYSP103
- FYSP104
- FYSP105
- FYSP106
- FYSP107
- FYSP110

Kurssin koodi: Anna kurssille koodi. Esim. FYSP100.

Opintokokonaisuus: Perusopinnot ▾

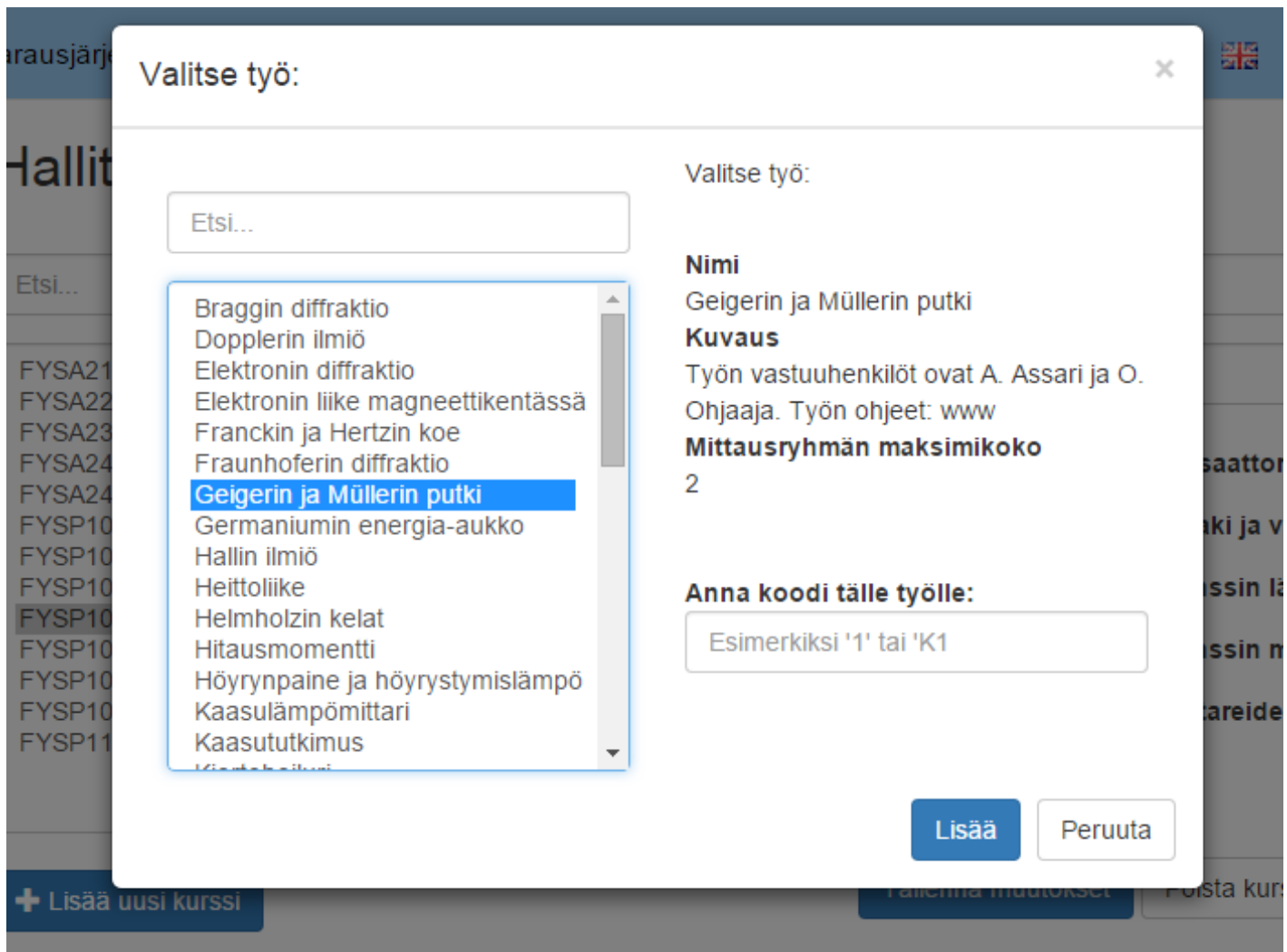
Työt: Kurssille ei ole määritetty töitä.

[+ Lisää työ](#)

[Tallenna muutokset](#) [Poista kurssi](#)

[+ Lisää uusi kurssi](#)

Kuva 3.19: Lisää uusi kurssi



Kuva 3.20: Työn lisääminen kurssiin

3.7.3 Työt

Työt-näkymässä listataan tietojärjestelmän sisältämät työt. Listattujen töiden tietoja voi muokata oikeassa reunassa ja painamalla Tallenna muutokset-painiketta. Vanhoja töitä pystyy poistamaan ja uusia töitä luomaan.

Työt vaativat käyttöönsä tietyn määrän resursseja. Kurssit -näkymässä tämän määrän voi määritellä ja töihin voi lisätä uusia resursseja. Työhön liittyviä resursseja voi poistaa muuttamalla työn kyseistä resurssia vaatiman määrän arvoksi nolla ja tallentamalla muutokset.

Kepler Laboratoriotöiden varausjärjestelmä 🇫🇮 🇬🇧 👤 Sulevi Superjuuseri ▾

Laboratoriotyöt

Etsi...

- Braggin diffraktio
- Dopplerin ilmiö
- Elektronin diffraktio
- Elektronin liike magneettikentässä
- Franckin ja Hertzin koe
- Fraunhoferin diffraktio
- Geigerin ja Müllerin putki
- Germaniumin energia-aukko
- Hallin ilmiö
- Heittoliike**
- Helmholtzin kelat
- Hitausmomentti
- Höyrynpaine ja höyrystymislämpö
- Kaasulämpömittari
- Kaasututkimus

+ Lisää uusi työ

Nimi (suomi): Heittoliike

Nimi (English): Ballistic motion

Kuvaus (suomi): Työn vastuuhenkilöt ovat A. Assari ja O. Ohjaaja. Työn ohjeet: www

Kuvaus (English): The supervisors in charge of the experiment are A. Assari and O. Ohjaaja. Experiment instructions: www

Osallittujia max

Resurssit: **Pyöriväalusta**

+ Lisää resurssi

Tallenna muutokset **Poista työ**

Kuva 3.21: Hallitse töitä

Kepler Laboratoriotöiden varausjärjestelmä 🇫🇮 🇬🇧 🔔 Sulevi Superjuuseri ▾

Ilmoitustaulu
Uusi varaus
Omat varaukset
Omat ryhmät
Ohjausvuorot
Hallitse vuoroja
Kurssit
Työt
Resurssit
Käyttäjäroolit
Muokkaa ilmoitustaulua

Laboratoriotyöt

Etsi...

- Braggin diffraktio
- Dopplerin ilmiö
- Elektronin diffraktio
- Elektronin liike magneettikentässä
- Franckin ja Hertzin koe
- Fraunhoferin diffraktio
- Geigerin ja Müllerin putki
- Germaniumin energia-aukko
- Hallin ilmiö
- Heittoliike
- Helmholzin kelat
- Hitausmomentti
- Höyrinpaine ja höyrystymislämpö
- Kaasulämpömittari
- Kaasututkimus

+ Lisää uusi työ

Nimi (suomi):

Nimi (English):

Kuvaus (suomi):

Kuvaus (English):

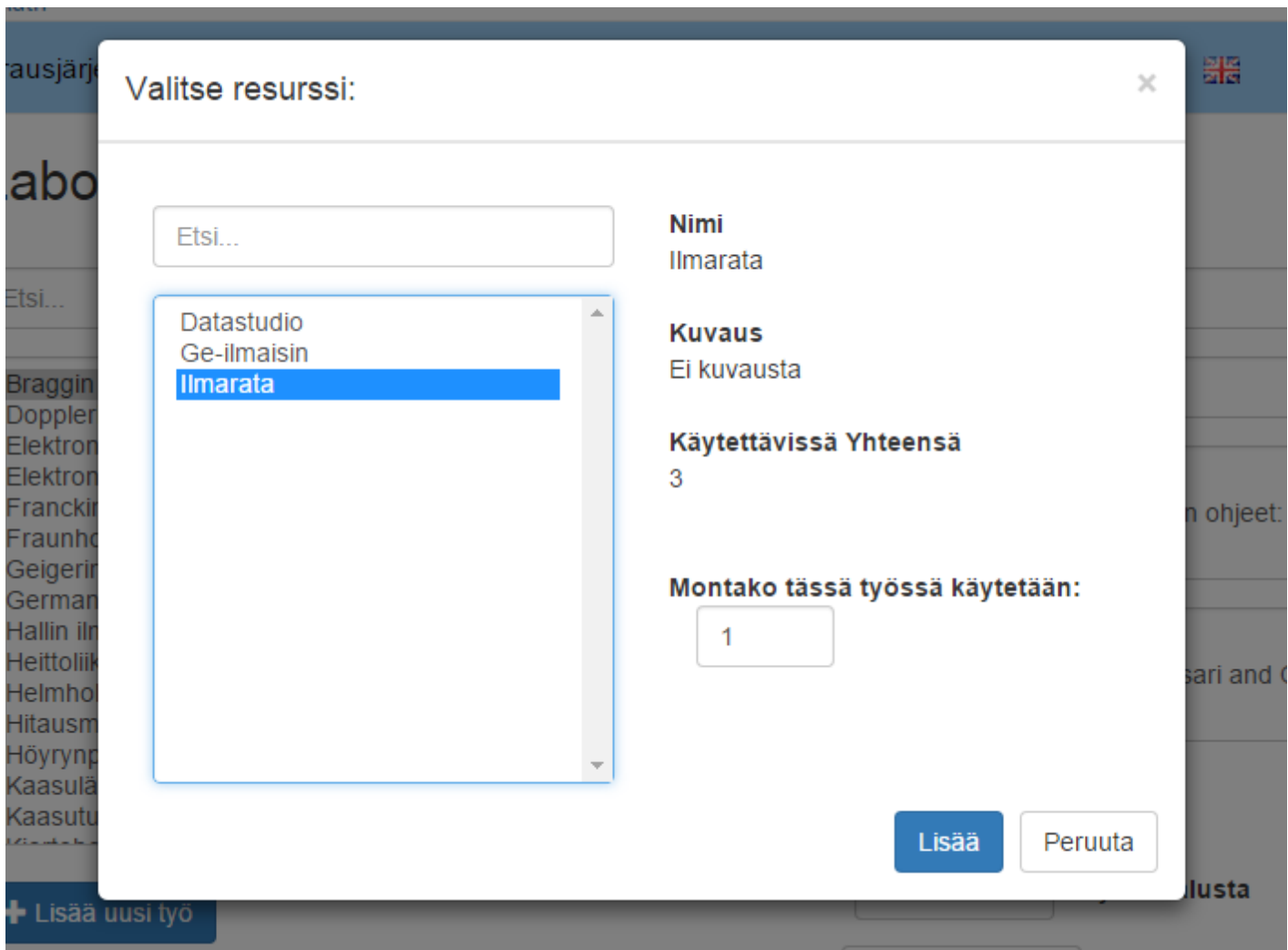
Osallittujia max

Resurssit: Työlle ei ole määritetty resursseja

+ Lisää resurssi

Tallenna muutokset Poista työ

Kuva 3.22: Lisää uusi työ



Kuva 3.23: Resurssi lisääminen työhön

3.7.4 Resurssit

Resurssit-näkymässä listataan tietojärjestelmästä löytyvät (töihin liittyvät) resurssit. Resurssien tietoja voi muokata oikealle olevista kentistä (ja muokatut tiedot tallennetaan käyttämällä Tallenna muutokset-painiketta). Lisäksi resursseista on olemassa tieto siitä, kuinka monta resurssia on missäkin tilassa. Myös kokonaan uusia resursseja voi lisätä. Resursseja ei voi poistaa järjestelmästä, mutta ne voi kirjata sellaisiin tiloihin, jossa niitä ei voi käyttää tai niiden määrän voi muuttaa nolleen.

Kepler Laboratoriotöiden varausjärjestelmä 🇫🇮 🇬🇧 🔔 Sulevi Superjuuseri ▾

Ilmoitustaulu
Uusi varaus
Omat varaukset
Omat ryhmät
Ohjausvuorot
Hallitse vuoroja
Kurssit
Työt
Resurssit
Käyttäjäroolit
Muokkaa ilmoitustaulua

Resurssit

Etsi...

- Datastudio
- Ge-ilmainen
- Ilmarata**
- Pyöriväalusta

+ Lisää uusi resurssi

Nimi (suomi):

Nimi (English):

Kuvaus (suomi):

Kuvaus (English):

Käytettävissä:

Pois käytöstä:

Rikki: **Yhteensä:**

Tallenna muutokset

Kuva 3.24: Hallitse resursseja

The screenshot shows the 'Resurssit' (Resources) section of the Kepler Laboratory reservation system. On the left is a navigation menu with options like 'Ilmoitustaulu', 'Uusi varaus', 'Omat varaukset', 'Omat ryhmät', 'Ohjausvuorot', 'Hallitse vuoroja', 'Kurssit', 'Työt', 'Resurssit' (highlighted), 'Käyttäjäroolit', and 'Muokkaa ilmoitustaulua'. The main content area is titled 'Resurssit' and features a search bar with the text 'Etsi...'. Below the search bar is a list of existing resources: 'Datastudio', 'Ge-ilmaisin', 'Ilmarata', and 'Pyöriväalusta'. A blue button labeled '+ Lisää uusi resurssi' is positioned below the list. To the right of the list is a form for adding a new resource, with fields for 'Nimi (suomi)', 'Nimi (English)', 'Kuvaus (suomi)', and 'Kuvaus (English)'. Below these are three input fields for 'Käytettävissä', 'Pois käytöstä', and 'Rikki', each containing the number '0'. A 'Yhteensä' field shows '0'. A blue button 'Tallenna muutokset' is at the bottom right of the form.

Kuva 3.25: Lisää uusi resurssi

3.7.5 Käyttäjäroolit

Käyttäjärooleihin liittyvän toiminnallisuuden yhdistäminen käyttöliittymään sovitin tilaajan kanssa jätettävän jatkokehitykseen. Nykyinen Käyttäjäroolit-näkymä on sekava ja koska se olisi syytä tehdä kokonaan uusiksi ennen käyttöönottoa, sitä ei esitetä.

3.7.6 Muokkaa ilmoitustaulua

Muokkaa ilmoitustaulua-näkymässä käyttäjä näkee ilmoitustaululla näkyvät ilmoitukset ja voi muokata niitä tai luoda kokonaan uusia.

Kepler Laboratoriotöiden varausjärjestelmä 🇫🇮 🇬🇧 ? Sulevi Superjuuseri ▾

Ilmoitustaulu

Uusi varaus

Omat varaukset

Omat ryhmät

Ohjausvuorot

Hallitse vuoroja

Kurssit

Työt

Resurssit

Käyttäjäroolit

Muokkaa ilmoitustaulua

Muokkaa ilmoitustaulua

Lisää viesti Näytä piilotetut

Lisää vuoroja tulossa! Sulevi Superjuuseri 13.6.2015 12.16

Laitoksella on ollut suunnitelmissa uusien vuorojen lisääminen laboratoriovuorojen kiertoon. Nyt asia on päätetty. Opiskelijoilta otetaan vastaan toiveita vuorojen ajankohdista.

Muokkaa **Pilota**

Muistakaa laskimenne! Sulevi Superjuuseri 13.6.2015 12.14

Muistakaa ottaa labroihiin mukaan laskimenne! Labroissa varalaskimia on tarjolla hyvin rajoitetusti.

Muokkaa **Pilota**

Kuva 3.26: Ilmoitusten hallinnointi

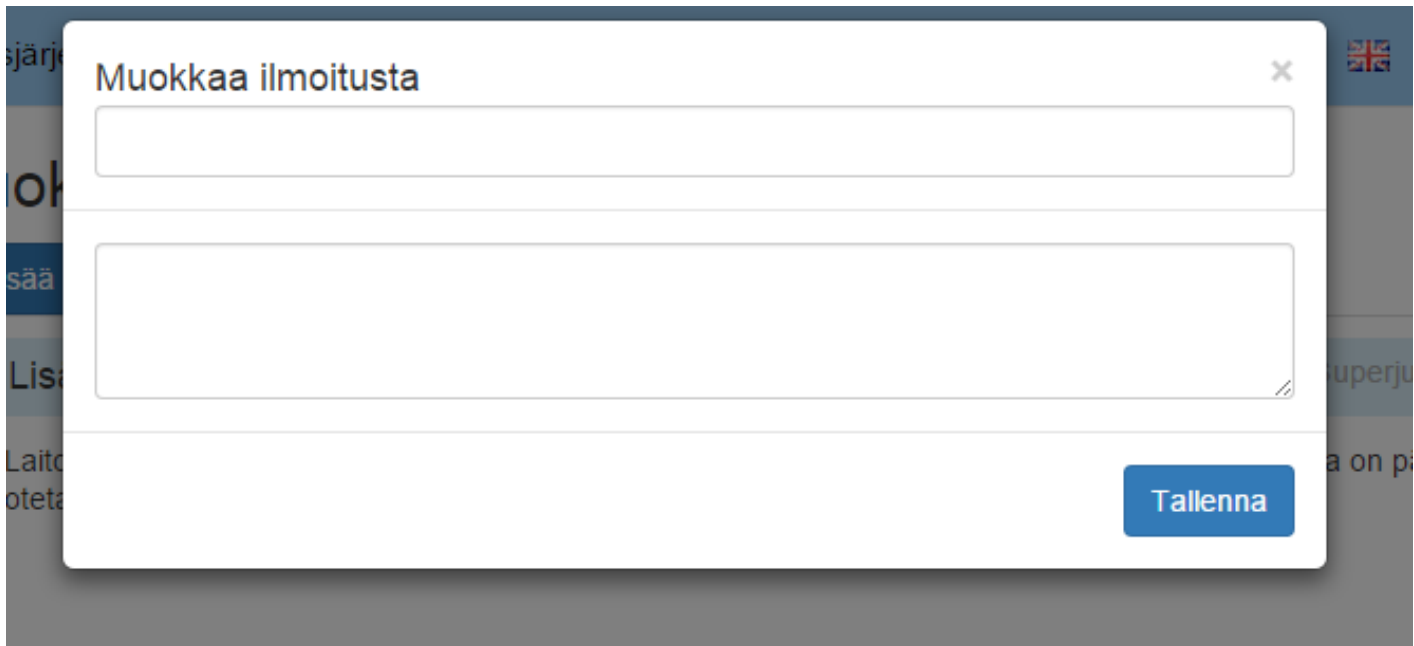
en varausjärj

Muokkaa ilmoitusta

Laitoksella on ollut suunnitelmissa uusien vuorojen lisääminen laboratoriovuorojen kiertoon. Nyt asia on päätetty. Opiskelijoilta otetaan vastaan toiveita vuorojen ajankohdista.

Tallenna

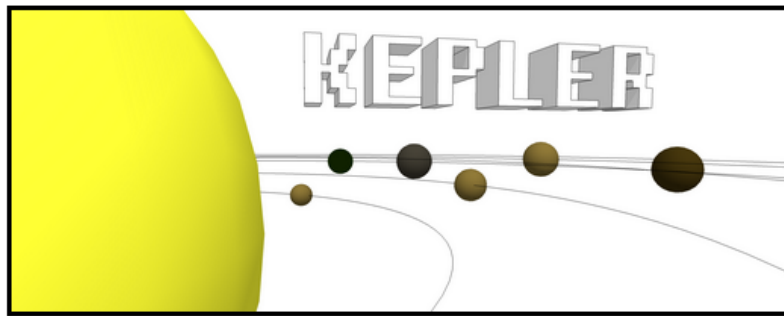
Kuva 3.27: Ilmoituksen muokkaaminen



Kuva 3.28: Lisää uusi ilmoitus

3.8 Kirjautumissivu

Käyttäjän syöttäessä selaimensa Kepler-järjestelmän osoitteen, hän saapuu kuvan 3.29 kirjautumissivulle, josta hän voi aloittaa järjestelmään sisäänkirjautumisen.



Kepler

Laboratoriovuorojen varausjärjestelmä

[Kirjaudu sisään](#)

Kuva 3.29: Kirjautumissivu

Tältä sivulta käyttäjä ohjataan Jyväskylän yliopiston OAuth-palveluun.

A screenshot of the login page for Jyväskylä University. The page has a blue header bar. Below the header, the text "Kirjautu sisään" is displayed in a large font. To the right of this text are two buttons: "fi" (selected) and "en". Below the header, there are two input fields: "Käyttäjätunnus" (Username) and "Salasana" (Password). Below the input fields is a blue button with the text "Kirjautu" and a right-pointing arrow.

Kuva 3.30: Jyväskylän yliopiston OAuth-sivu

Kirjautumisen jälkeen OAuth-palvelu pyytää lupaa antaa Kepler-järjestelmän käyttää joitain tiettyjä palveluita. Ilman näitä lupia, Kepler ei toimi.



Lupapyyntö

fi en

Palvelu Kepler-sovelluksen paikallinen palvelin haluaa lupia seuraaviin oikeuksiin, sallitaanko?

- call_current_user_info
- call_korppi-ical

Kyllä En

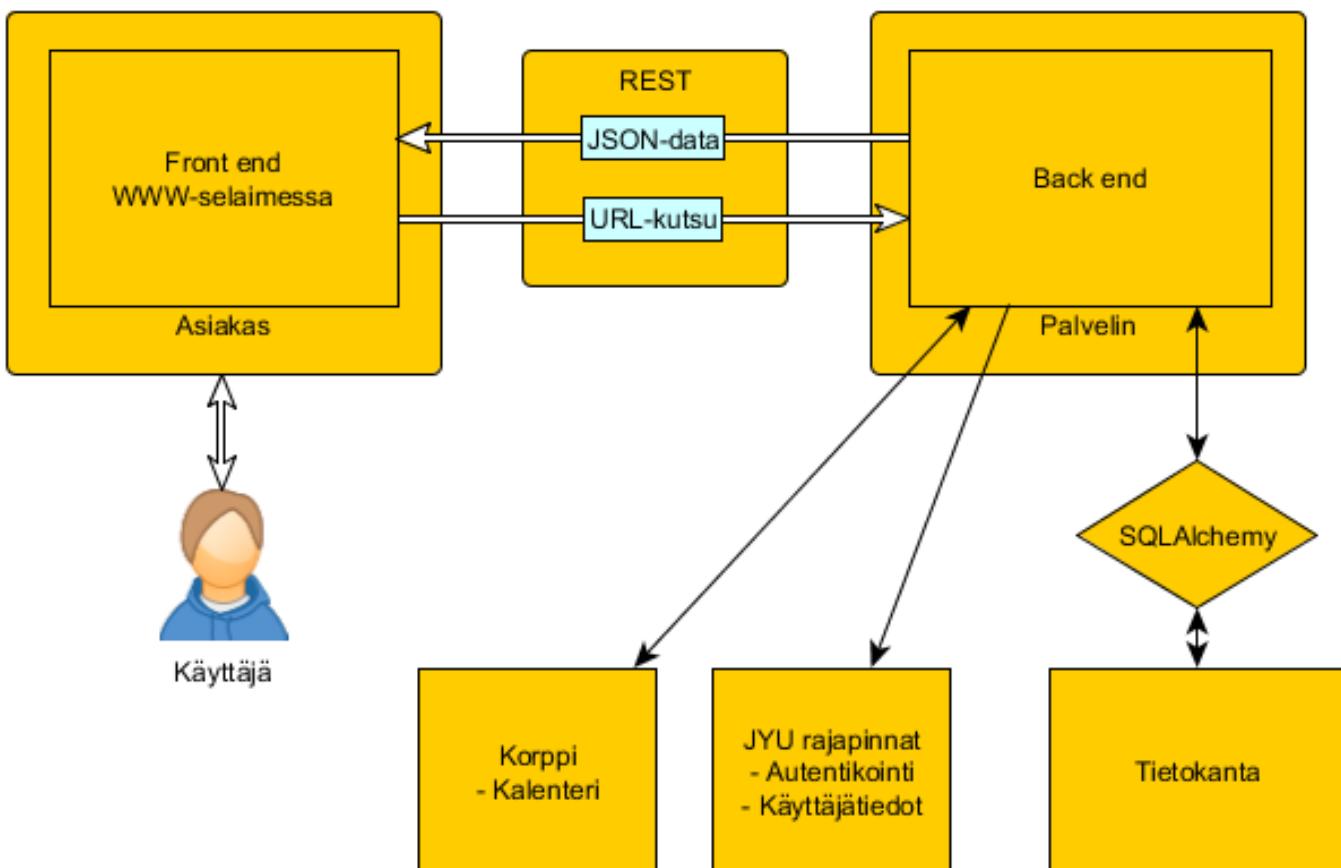
Kuva 3.31: Puuttuu, koska testi OAuth ei toimi Rädyn tunnuksilla kirjoittamisen hetkellä

4 Sovelluksen kokonaisrakenne ja rajapinnat

Luvussa kuvataan tietojärjestelmän kokonaisrakennetta, käytettyjä ulkopuolisia komponentteja ?? sekä käytettyjä ulkopuolisia rajapintoja ?. Kepler-projektissa kehitetyn käyttöliittymän käyttäjille tarjoamat tiedot ja toiminnot on kuvattu tarkemmin vaatimusmäärittelyssä [7]. Sovelluksen luokat ja niiden rajapinnat kuvataan yksityiskohtaisesti luokkadokumentaatioissa [?] ja [?].

4.1 Sovelluksen kokonaisrakenne

Sovelluksen kokonaisrakennetta ja rajapintoja on kuvattu kuvassa 4.1.



Kuva 4.1: Sovelluksen kokonaisrakenne

Sovellus jakautuu seuraaviin kokonaisuuksiin:

Front end	on WWW-selaimessa pyörivä osa sovelluksesta, eli Keplerin käyttöliittymä.
REST	on REST-periaatteita noudattava rajapinta, jonka välityksellä back end ja front end keskustelevat.
Back end	on palvelimella pyörivä osa sovelluksesta, joka suorittaa toiminnot ja tekee toimintoihin liittyvät tarkistukset.
SQLAlchemy	on kolmannen osapuolen komponentti, joka pitää huolen back endin ja tietokannan välisestä kommunikaatiosta.
Tietokanta	sisältää tietojärjestelmän varastoitamat ja tarvitsevat tiedot.
Korppi-kalenteri	sisältää Jyväskylän yliopiston opiskelijoiden ja henkilökunnan kalentritietoja.
JYU-rajapinnat	ovat Jyväskylän yliopiston OAuth-palvelun tarjoamia rajapintoja, jotka mahdollistavat käyttäjän autentikoinnin ja käyttäjätietojen noutamisen.

Kuvassa 4.1 esitettävistä kokonaisuuksista Korppi-kalenteri ei toteutunut Kepler-projektin puitteissa. Lisätietoa tästä löytyy luvusta ??.

Sovelluksen yleisessä käyttötapauksessa toiminta alkaa siitä, että käyttäjä navigoi selaimellaan haluamaansa näkymään. Tämän jälkeen hän suorittaa selaimellaan jonkinlaisen toiminnon, jolloin järjestelmän palvelimelle lähtee HTTP POST-pyyntö,

jonka body osassa on mahdollisesti joitain parametreja JSON-muodossa. Palvelin suorittaa käyttäjän pyytämän toiminnon, mikäli se on sallittua tehdä. Palvelin joutuu todennäköisesti käsittelemään tietokannassa olevaa tietoa suorittaakseen toiminnon. Tämän jälkeen palvelin lähettää käyttäjän selaimelle vastauksen JSON-muodossa. Käyttäjän selaimessa pyörivä Kepler-järjestelmän JavaScript käsittelee saamansa JSON:in ja esittää sen sisällön käyttäjän selaimessa.

4.2 Ulkopuoliset komponentit

Tietojärjestelmä hyödyntää ulkopuolisia komponentteja seuraavista kirjastoista ja ohjelmista:

Beaker	on Python WWW-sovelluksia tukeva kirjasto sessionhallintaan. Sen ovat tehneet Ben Bangert, Mike Bayer ja Philip Jenvey.
Bootstrap	on Mark Otton ja Jacob Thorntonin kehittämä HTML-, CSS- ja JavaScript-kieliä tukeva kirjasto skaalautuvien WWW-sivujen luontiin.
dateutil	on Pythonin omaa <code>DateTime</code> -tyyppiä laajentava kirjasto. Sen on tehnyt Yaron de Leeuw.
docutils	on Python-kirjasto, joka muuttaa dokumentaatiota suosittuihin formaatteihin (esimerkiksi HTML, XML ja \LaTeX). Sen on kehittänyt "docutils-develop list".
FullCalendar	on jQuery:n laajennus, jolla voi helposti luoda ja muokata WWW-sivulla esitettävää kalenteria. Sen on kehittänyt Adam Shaw.
jQuery	on alun perin John Resigin kehittämä JavaScript-kirjasto, joka laajentaa JavaScriptin perustoiminnallisuuksia ja helpottaa niiden käyttöä.
icalendar	on Python-kirjasto iCalendar-formaatissa olevan (kalenteri)datan lukemiseen, käsittelyyn ja luomiseen. Sen on kehittänyt Plone Foundation.

Pyramid	on Pylons Projectin tarjoama Python WWW-sovelluskehys, jonka päälle Kepler-järjestelmä on rakennettu.
pyramid_beaker	on Pylons Projektin kehittämä Python-kirjasto, jolla Beakerin saa helposti integroitua Pyramidiin. Kirjasto ei ole enää aktiivisessa kehityksessä.
pyramid_mailer	on Pylons Projectin kehittämä Python-kirjasto, jonka avulla Pyramid:lla voi helposti lähettää sähköpostiviestejä.
Requests	on Python kirjasto HTTP- ja HTTPS-pyyntöjen tekemiseen. Se on tehty Pythonin omaa kirjastoa (urllib2) helpommin käytettäväksi. Sen on kehittänyt Kenneth Reitz.
SQLAlchemy	on alun perin Michael Bayerin kirjoittama Python-kirjasto, jolla pystytään käyttämään relaatiotietokantoja helposti.

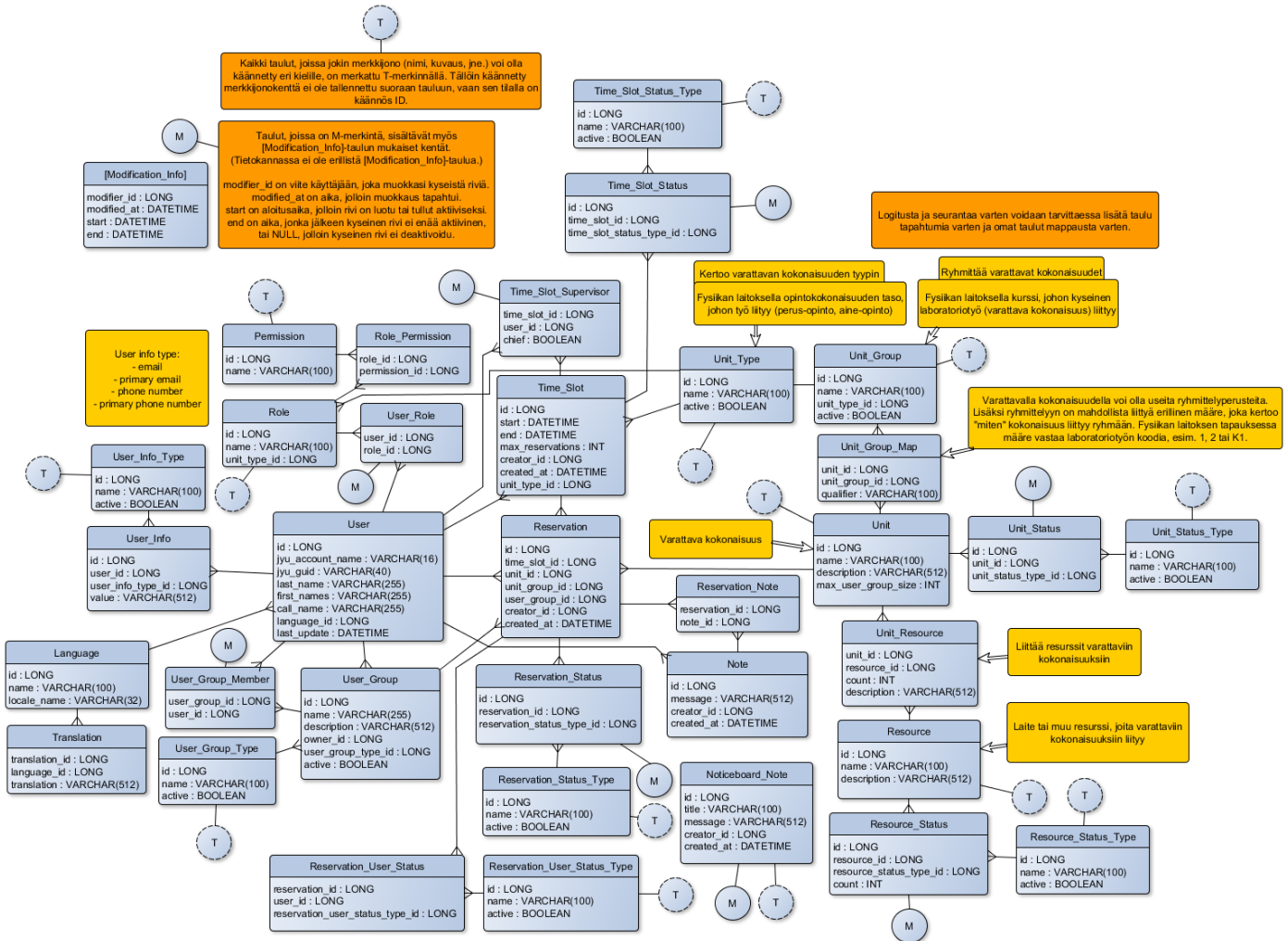
4.3 Olennaisimmat muutokset toteutusratkaisuihin

Sovelluksen kokonaisrakenteeseen ei tehty muutoksia projektin aikana. Suurimmat muutokset olivat sovelluksen sisäiseen toimintalogiikkaan liittyvät refaktoroinnit. Nämä refaktoroinnit koostuivat usein käytettyjen toimintojen muuttamisesta käyttämään erinäisiä apuluokkia.

Refaktorointi tapahtui esimerkiksi virheviestien esittämisen yhteydessä, kun ne muutettiin pelkistä merkkijonoista käyttämään erilaisia apuluokkia, jolloin viesteistä sää yhdenmukaisia. Virheviestien esittäminen refaktorointiin Kepler-projektin aikana pariinkin kertaan ja nyt virheviestit voi (pääasiassa) esittää helposti ja yhtenevästi käyttöliittymässä.

4.4 Tietokannan rakenne

Koska tietojärjestelmästä oli tavoitteena kehittää yleiskäyttöinen, piti tietokannastakin tehdä sellainen.



Kuva 4.2: Tietokannan rakenne

Tietokannassa toistuu rakenne, jossa "perustaululle" on olemassa myös Status- ja Type-taulut. Status taulu kuvaa aina perustaulun elementin jonkinlaista tilaa. Type-tauku taas antaa Status-taulun kuvaamalle tilalle nimen. Status- ja Type-taulut on eritelty erillisiksi rakenteiksi sen takia, että eri tilojen nimiä (ja mahdollisesti muita ominaisuuksia, mikäli sellaisia tulevaisuudessa tulee) olisi helppo muuttaa hajottamatta mitään olemassa olevaa toimintalogiikkaa.

Tietokannan kuvassa 4.2 on käytetty merkintöjä T ja M, kuvan selkeyttämiseksi (samoin kuin monia taulujen välisiä suhteita on jätetty merkitsemättä). Merkintä M tarkoittaa, että tauluun sisältyy kentät `modifier_id`, `modified_at`, `start` ja `end`, kuten kuvassa 4.2 näkyvässä [ModificationInfo] taulussa näkyy. Merkintä T taas puolestaan tarkoittaa, että merkinnän T omaavassa taulussa ei oikeasti säilötäkään merkkijono(j)a, vaan niiden sijaan säilötään `translation_id`, joka viittaa haluttuun merkkijonoon.

Koska tietokantaa voi olla vaikea ymmärtää, käydään jokaisen taulun tarkoitus läpi alla. Taulut käydään läpi kokonaisuuksittain. Kuvassa 4.2 olevat alaviivat jätetään toistamatta, koska ne eivät ole käytössä tietojärjestelmän toteutuksessakaan.

Permission	sisältää eri toimintojen vaatimat valtuudet.
RolePermission	liittää valtuudet rooleihin.
Role	sisältää tietojärjestelmän eri käyttäjäroolit.
UserRole	sisältää tiedon siitä, mitä rooleja kullakin käyttäjällä on.
User	sisältää tietoja tietojärjestelmän käyttäjistä. <code>language_id</code> -kenttä kertoo käyttäjän oletuskielivalinnan. <code>last_update</code> -kenttää ei nykyisellään käytetä.
UserInfo	sisältää käyttäjään liitettävää tietoa. Sen merkittävien ero <code>User</code> -tauluun on se, että sen sisältämät tiedot ovat luonteeltaan sellaisia, jotka muuttuvat useammin. Nykyisellään <code>UserInfo</code> -taulu sisältää vain käyttäjän yhteystietoja.
UserInfoType	sisältää tiedon <code>UserInfo</code> -taulun rivi sisältämän tiedon tyypistä.
Language	sisältää tietoa järjestelmän tukemista kielistä (ja ne tunnistavista <code>localeista</code>).
Translation	sisältää kaikki tietojärjestelmän sisältämän tiedon ja virheviestien käännökset.

UserGroup	sisältää tietoa järjestelmässä olevista käyttäjien muodostamista ryhmistä.
UserGroupMember	sisältää kuhunkin ryhmään kuuluvat käyttäjät.
UserGroupType	sisältää tiedon ryhmän tyypistä (esimerkiksi, henkilökohtainen ryhmä tai opiskelijoiden ryhmä).
TimeSlot	sisältää tietoa järjestelmässä olevista ajanjaksoista, joiden ajalle voi tehdä varauksia.
TimeSlotSupervisor	sisältää tiedon ajanjaksojen valvojana tai ohjaajana toimivista henkilöistä. Näistä henkilöistä tiedetään myös, kuka tai ketkä heistä ovat vastuussa.
TimeSlotStatus	liittää yhteen ajanjakson ja siihen liittyvän tilan.
TimeSlotStatusType	sisältää tilat, joissa ajanjakso voi olla (esimerkiksi suunnitteilla oleva tai varmistettu). Taulun sisältöä ei oteta nykyisen toteutuksen toimintalogiikassa huomioon.
Reservation	sisältää tiedon järjestelmän ajanjaksoille tehdyistä varauksista ja niihin liittyvistä tiedoista.
ReservationStatus	liittää yhteen varauksen ja siihen liittyvän tilan.
ReservationStatusType	sisältää tilat, joissa varaus voi olla (esimerkiksi pääkäyttäjän peruma tai aktiivinen).
ReservationUserStatus	liittää yhteen varauksen ja siihen liittyvän käyttäjän tilan.
ReservationUserStatusType	sisältää varaukseen osallistuneen henkilön mahdolliset tilat (esimerkiksi paikalla tai poissa).
ReservationNote	liittää yhteen varauksen ja siihen liittyvän viestin.
Note	sisältää kaikki järjestelmään kirjatut viestit.
NoticeboardNote	sisältää tiedon ilmoitustaulun ilmoituksista ja niihin liittyvistä tiedoista.

Unit	sisältää varattavat yksiköt ja niihin liittyvät tiedot (esimerkki tällaisesta yksiköstä on fysiikan laboriotyö). Varattavat yksiköt muodostuvat resursseista.
UnitGroupMap	liittää yhteen varattavat yksiköt ja niiden muodostamat yksiköiden ryhmät. Tämän lisäksi taulu sisältää varattavan yksikön ja siihen liittyvän kokonaisuuden yksilöivän tunnisteiden.
UnitGroup	sisältää tiedon siitä, mihin varattavien yksiköiden kokonaisuuteen (esimerkiksi kurssin vaatimat laboriotyöt) mikäkin yksikkö kuuluu.
UnitType	sisältää tiedon varattavan yksikön tyypistä (esimerkiksi perus- tai aineopinnot).
UnitStatus	liittää yhteen varattavan yksikön ja siihen liittyvän tilan.
UnitStatusType	sisältää tilat, joissa varattava yksiköt voivat olla. Taulun sisältöä ei oteta nykyisellään huomioon järjestelmän toimintalogiikassa.
UnitResource	liittää resurssit varattaviin yksiköihin (esimerkiksi mittanauhat laboriotyöhön).
Resource	sisältää tietoa järjestelmässä olevista resursseista (esimerkiksi mittanauhat), joista muodostetaan varattavia yksiköitä.
ResourceStatus	liittää resurssit ja niihin liittyvän tilan. Taulu myös sisältää tiedon siitä, kuinka monta resurssia kussakin tilassa on.
ResourceStatusType	sisältää tilat, joissa resurssi voi olla.

4.5 Muutettavat asetukset

Käyttäjä voi muuttaa kielivalintaansa pysyvästi omista tiedoistaan, kuten luvussa 3.2.5 kuvattiin.

Keplerin ylläpitäjä voi muuttaa joitakin asetuksia, muuttamalla tiedostoa `config.py` (projektin juuresta katsoen sen tiedostopolku on `/kepler/config.py`). Alla on kuvattuna esimerkki sen sisällöstä.

```
class Config(object):
    """Config class defines the configuration values of the
    Kepler system."""

    time_slot_max_reservations_default = 6
    """The default value for the limit of reservations that
    can be made on a time slot."""

    reservation_cancel_workdays_limit = 2
    """Reservations can be canceled only if there is more than
    the specified number of workdays before the start of the
    time slot."""

    reservation_make_workdays_limit = 2
    """Reservations can be made only on time slots that are
    more than the specified workdays in the future."""

    user_group_max_reservations = 4
    """One user group can have at most the specified number of
    active reservations."""

    user_group_min_size = 2
    """Defines the minimum number of members a user group can
    have."""

    user_group_max_size = 2
    """Defines the maximum number of members a user group can
    have."""

    default_language_id = constants.Language.finnish_id.value
    """The ID of the default language of the system."""
```

Lisäksi on olemassa asetustiedostot `development.ini` ja `production.ini` (jotka löytyvät järjestelmän kansiorakenteen juuresta), mutta näissä ei kannata juuri-

kaan muuttaa sähköpostiasetuksia lukuun ottamatta. Nämä asetukset astuvat voimaan, kun palvelu käynnistyy. `development.ini` on tarkoitettu kehittämiskäyttöön eikä sitä saa missään nimessä käyttää tuotantoympäristössä! Alla on esitetty `production.ini`:n nykyinen sisältö.

```
###
# app configuration
# http://docs.pylonsproject.org/projects/pyramid/en/1.5-branch/narr/
# environment.html
###

[app:main]
use = egg:Kepler

pyramid.reload_templates = false
pyramid.debug_authorization = false
pyramid.debug_notfound = false
pyramid.debug_routematch = false
pyramid.default_locale_name = en
pyramid.includes =
    pyramid_tm
    pyramid_chameleon
    pyramid_beaker
    pyramid_mailer

sqlalchemy.url = sqlite:///%(here)s/Kepler.sqlite

###
# Beaker session config
###

# The session invalidation timeout in seconds. (900s = 15min)
session.timeout = 900
# Tells web browser that the session ID cookie is to be sent only over
# secure connection.
session.secure = true
# Used with the HMAC to ensure session integrity.
```

```
session.secret = 'b1203525208e0b0934b43aeb0c5ccf00ee832071'
```

```
[server:main]
use = egg:waitress#main
host = 127.0.0.1
port = 6543
# Add trusted_proxy IP to allow X_FORWARDED_PROTO header to
# override default URL scheme
# 127.0.0.1 = IP of localhost
trusted_proxy = 127.0.0.1
# url scheme when using secure HTTPS connection :
url_scheme = 'https'
```

```
###
# logging configuration
# http://docs.pylonsproject.org/projects/pyramid/en/1.5-branch/narr/logging
###
```

```
[loggers]
keys = root, kepler, sqlalchemy
```

```
[handlers]
keys = console
```

```
[formatters]
keys = generic
```

```
[logger_root]
level = WARN
handlers = console
```

```
[logger_kepler]
level = WARN
handlers =
qualname = kepler
```

```
[logger_sqlalchemy]
level = WARN
handlers =
qualname = sqlalchemy.engine
# "level = INFO" logs SQL queries.
# "level = DEBUG" logs SQL queries and results.
# "level = WARN" logs neither. (Recommended for production systems.)

[handler_console]
class = StreamHandler
args = (sys.stderr,)
level = NOTSET
formatter = generic

[formatter_generic]
format = %(asctime)s %(levelname)-5.5s [% (name)s] [% (threadName)s] % (m
```

4.6 Ulkoiset rajapinnat

Kepler-projektissa käytetään Jyväskylän yliopiston tietojärjestelmien rajapintoja.

4.6.1 Korppi-kalenteri

Alun perin suunnitelmissa oli tuoda Kepler-järjestelmän kalentereihin Korppi-kalenterin tapahtumat ja viedä Kepler-järjestelmän tapahtumat Korppi-kalenteriin. Tämä jäi kuitenkin toteuttamatta yhteistyön Korpin kanssa viivästyessä. Korppi-kehittäjät tekivät OAuthin kautta toimivan esimerkkitoteutuksen, mutta varsinainen toteutus jäi vielä tulevaisuuteen.

Keplerissä on lähdetty työstämään esimerkkitoteutuksen mallidatan (iCalendar muotoista) parsimista ja käyttöä. HTTP-kutsuihin käytetty Requests-kirjasto ei kuitenkaan toimi kyseisessä tapauksessa odotetulla tavalla ja kyseinen ominaisuus pitäisi toteuttaa Pythonin valmiilla urllib2-kirjastolla. Koska ongelma ilmeni myöhäisessä vaiheessa projektia, se sovittiin tilaajan kanssa jätettävän jatkokehitykseen. iCalendar muotoisen datan parsiminen ja tuottaminen kyllä toimivat.

4.6.2 OAuth

Kepler-järjestelmään kirjaudutaan Jyväskylän yliopiston tarjoaman OAuth2-rajapinnan kautta. Näin Kepler-järjestelmän ei itse tarvita huolehtia autentikoinnista.

OAuth-rajapinnan kautta Kepler myös saa käyttäjistä tietoa (esimerkiksi nimet ja sähköpostiosoitteen), joten käyttäjän ei tarvitse erikseen täyttää ja päivittää tietojaan Kepler-järjestelmään.

Jyväskylän yliopiston OAuth-rajapinnasta löytyy tarkempaa tietoa sen ohjeista (<https://dev-docs.app.jyu.fi/>) **??**. Tämän lisäksi kannattaa tutustua OAuthin viralliseen RFC:hen (<https://tools.ietf.org/html/rfc6749>) **??**.

Kepler-järjestelmään OAuth-autentikoinnin ja tietojen haun toteutus on tehty itse, sillä mikään löydetyistä valmiista (ja yhä aktiivisesti päivitetyistä) kirjastoista ei olisi toiminut ilman muutoksia. Jyväskylän yliopiston OAuth-rajapinnalla on erikoisvaatimus, jota RFC:ssä ei ole. Jyväskylän yliopiston OAuth-rajapint vaatii, että `authorization code grant flow`:ta käytettäessä `client_id` sisällytetään POST-pyyntöön, kun rajapinnalta pyydetä `tokenia`. Kepler-järjestelmä käyttää OAuth-autentikoitiin `authorization code grant flow`:ta.

5 Tavoitteiden toteutuminen

Luvussa kuvataan vaatimusten toteutumisesta sekä heikkoja ja puutteellisia toteutusratkaisuja. Sovelluksen vaatimukset ja niiden toteutuminen on kuvattu vaatimusmäärittelyssä [7].

Vaatimusmäärittelyn ensimmäisen prioriteetin 38:sta vaatimuksesta toteutui kokonaan 32. Yhtä vaatimusta (8.2.4 50 yhtäaikaisen käyttäjän palveleminen) ei testattu, joten se lasketaan toteutumattomaksi.

Rajallisen aikataulun vuoksi sovellukseen jäi myös heikkoja ja puutteellisia toteutusratkaisuja. Ohjelmointikäytännöt ja niiden toteutuminen on kuvattu projektiraportin [5] luvussa 5.5.

5.1 Vaatimusten toteutuminen

Vaatimusten toteutuminen vaatimuskohtaisesti on kuvattu vaatimusmäärittelyssä [7]. Pakollisista 38 vaatimuksesta toteutui kokonaan 32 vaatimusta ja kaksi toteutettiin osittain. Osittain toteutettu pakollinen vaatimus 7.7.9 liittyy tarkastukseen laboratoriotyötä (tietokannassa `Unit`-taulussa) poistettaessa, silloin kun se kuuluu kurssiin (tietokannassa `UnitGroup`-taulussa). Osittain toteutettu pakollinen vaatimus 8.2.3 liittyy järjestelmään vastustuskykyyn CSRF-hyökkäystä (cross site request forgery) vastaan, sillä tätä vastaan on varauduttu ainakin paikoittain, mutta järjestelmällisesti sitä ei ole testattu.

Tärkeistä 27 vaatimuksesta toteutettiin 15. Osittain toteutettiin seuraavat vaatimukset:

- | | |
|--------------|---|
| 7.5.4 | Käyttäjä pystyy lisäämään käynnöksen ilmoituksen otsikolle ja sisällölle. (Tämän pystyy tekemään, muttei luontevalla tavalla. Tämän takia sitä pidetään osittain toteutettuna.) |
| 7.9.1 | (Pää)käyttäjä pystyy lisäämään käyttäjille uusia rooleja. Toteuttuna testamattomana palvelinkoodissa. |

- 7.9.2** Käyttäjä pystyy merkitsemään käyttäjän roolin poistetuksi. Toteuttuna testaamattomana palvelinkoodissa.
- 8.1.4** Kalenterinäkymän tuontiin Korpista käytetään iCalendar-muotoa. Valittu toteutusratkaisu ei toimi. Sen saisi toimimaan, kun kirjoittaisi sen uudelleen käyttäen urllib2:ta Requestsin sijaan.

Mahdollisista 40 vaatimuksesta toteutettiin 19 ja osittain toteutettiin 5 vaatimusta. Toteuttamatta jäi 16 mahdollista vaatimusta. 20 ideatason vaatimusta sovittiin jo alussa projektin ulkopuolelle. Vaatimusmäärittelyyn ei kirjattu prioriteetilla Ei toteuteta yhtään vaatimusta.

Vaatimusten toteutumisesta kerrotaan myös vaatimusmäärittelyn [7] luvussa 9.

5.2 Suoritetut testaukset ja niiden tulokset

Sovelluksen toteutusvaiheessa jokainen ryhmän jäsen suoritti manuaalista testausta, testaten tekemiään muutoksia. Testausta vaikeutti se, että käyttöliittymän ja palvelimen toteutus riippuivat olennaisesti toisistaan, jolloin molempien osien piti olla valmiina testausta varten. Näin ollen esimerkiksi palvelinkoodin virheitä ei yleensä huomattu ennen kuin sitä käyttävä käyttöliittymän osa valmistui.

Projektin lopussa Mikko Kuhno laati testausraportin [?], joka kattoi suoritetut testitapaukset. Järjestelmätestaus suoritettiin Windows-ympäristössä Internet Explorer (versio 11.0.9600.17801) ja Mozilla Firefox (versio 35.0.1) selaimilla. Testauskertoja ei suoritettu todellisessa käyttötilanteessa.

Projektin aikana järjestettiin käytettävyytestauspäivä, jonka aikana käytettävyyshuoltaja Johanna Silvennoinen antoi palautetta sovelluksen käytettävyydestä.

Projektin loppupuolella Anu Koskela suunnitteli ja suoritti järjestelmän käytettävyyshuollon. Käytettävyyshuollon testauksessa käyttäjinä olivat Jyväskylän fyysikan laitoksen opiskelijoita, joista osa toimii laboratoriovuorojen ohjaajina ja osa oli tavallisia opiskelijoita. Käytettävyyshuollon testauksessa havaitut ongelmat huomioitiin sovelluksen toteutuksessa. Käytettävyyshuollon testauksesta tehtiin suunnitelma [?] ja muistio [2].

Projektin aikana sovelluksen eri versioita koekäyttivät vastaava ohjaaja Jukka-Pekka Santanen, tekninen ohjaaja Petri Partanen, sekä tilaajan edustajat Sakari Juutinen ja Panu Rahkila. Erityisesti Jukka-Pekka Santanen ja tilaajan edustajat antoivat sovelluksesta palautetta.

5.3 Heikot ja puutteelliset toteutusratkaisut

Käyttöliittymän JavaScriptissä syötetään paikoittain sivulle suoraan HTML-koodia. Tämä pitäisi korvata oikeaoppisesti JavaScriptillä.

Käyttöliittymässä käytetään vielä paikoin JavaScriptin alertteja ilmoitusten yhteydessä. Nämä olisi syytä korvata viher- ja punataustaisilla teksteillä (niin kuin monessa kohtaa on jo tehtykin).

Kun selainikkunaa pienennetään (tai Kepleriä käytetään laitteella, jonka resoluutio on pieni), järjestelmän nimen yhteydessä oleva järjestelmän kuvaus ei rivity oikein ja jää osittain navigointipalkin alle piiloon.

Järjestelmässä pitäisi käyttää kaikkialla yhtenevää väriteemaa. Nykyisellään asiaa ei ole perinpohjaisesti selvitetty, joten värit voivat oli eri näkymissä eriäviä.

Kun käyttäjän sessio vanhenee (ja hänet kirjataan automaattisesti ulos), pitäisi tilanteesta antaa käyttäjälle selkeä ilmoitus hänen yrittäessään tehdä jotain (pelkän kirjautumisnäkymän uudelleenohjaamisen sijasta).

HTML:ään tulisi lisätä selitteet (eli lähinnä titlet) erilaisia lukijalaitteita varten, jotta Kepler olisi käytettävämpi erikoistapauksissa (esimerkiksi sokealle ihmiselle).

Uutta vuoroa luotaessa pitäisi pystyä määrittelemään, onko vuoro vasta suunnitteilla vai tuleeko se toteutumaan (nyt käyttöliittymässä on valinta, jolla ei ole vaikutusta lopputulokseen).

Käyttäjäroolienhallinta on toteutettuna palvelimen päässä (vaikkakin testaamattomana), mutta sille pitäisi vielä tehdä käyttöliittymä. Sitä varten oleva käyttöliittymä ei nykyisellään sisällä mitään toiminnallisuutta eikä se ole käytettävyydeltään hyvä (joten se pitäisi tehdä kokonaan uudestaan).

Palvelimen koodissa luotetaan siihen, että tietokannasta tuleva data ei aiheuta tietoturvaongelmia. Käyttäjän syötettä ei kyllä pitäisi päästä käsittelemättömänä tieto-

kantaan, mutta tietokannasta tulevaan dataan ei silti ole viisasta sokeasti luottaa ja sillekin olisi syytä suorittaa jonkinlaisia operaatiota.

Järjestelmä ei tee minkäänlaisia lokeja WWW-palvelimen omien lokien lisäksi. Tämä olisi syytä korjata. Jos lokit haluaa liittää tietokantaan, voisi ne toteuttaa esimerkiksi samantyyppisellä ratkaisulla kuin millä merkkijonojen käännökset on nykyisessä järjestelmässä toteutettu (Translation-taulu).

Järjestelmä ei pidä yllä kovin kattavia historiatietoja. Joistain tauluista ei löydy minkäänlaisia historiatietoja. Niistä tauluista, joista historiatietoja löytyy, koskettavat ne vain viimeisintä muutosta. Kattavammissa historiatiedoista olisi hyötyä ylläpidolle ja niistä voisi tehdä tilastoja (esimerkiksi siitä, kuinka moni vuorolle ilmoittaunut sinne keskimäärin saapuu).

OAuth-toteutuksen virheenkäsittely ei ole kovin sulavaa tai kattavaa. Virheviestejä ei myöskään lähetetä muulle järjestelmälle yhtenevällä tavalla.

Sähköpostiviestit tulisi testata sähköpostipalvelimen kanssa (järjestelmää kehittäessä niitä on testattu vain paikallisesti).

Varattavien töiden (varattavien yksiköiden, tietokannassa Unit-taulu), resurssien (tietokannassa Resource-taulu) ja kurssien (tietokannassa UnitGroup-taulu) poiston yhteydestä puuttuvat kaikki mielekkäät tarkistukset.

Nykyinen ulkopuolisen iCalendar-syötteen lukemisen toteutus ei toimi. Se pitäisi kirjoittaa uusiksi käyttäen urllib2:ta Requestsin sijaan (muilta osin se toimii).

Yksi suurimmista Kepler-järjestelmän koodin ongelmista on se, että etenkin palvelinkoodissa toistetaan samoja kaavoja hyvinkin paljon. Eri toteutusten tarpeissa on kuitenkin tarpeeksi eroja, että emme nähneet järkeväksi ajankäytöksi tehdä Kepler-projektin puitteissa riittävän yleisen tason apuluokkia koodin modularisuuden parantamiseksi.

Etenkin erilaisten palvelinkoodin toimintalogiikkaan liittyvien tarkastusten yhteydessä olisi syytä käyttää luettavuuden nimissä aliohjelmia. Nykyisellään koodi voi olla raskasta luettavaa, ennen kuin sitä oppii lukemaan (sillä samantyyppinen koodi toistuu toiminnosta toiseen).

Pääkäyttäjän näkymissä ei ole käytettävyyteen panostettu yhtä paljon kuin muissa näkymissä (ajan puutteen ja priorisoinnin vuoksi).

Nykyisellään kirjautumisnäkyvä on askeettinen, joten pieni ulkonäön parantelu olisikin varmaan paikallaan ennen järjestelmän käyttöönottoa.

Tietokannan alustuskripti syöttää tietokantaan aina myös mallidataa, eli se ei sovi tuotantoympäristöön. Siitä kuitenkin saisi pienellä vaivalla poistettua tämän mallidatan syötön (mallidatan syöttävä versio kannattaa kuitenkin säilyttää kehityskäyttöä varten).

5.4 Sovelluksen yleiset jatkokehitysideat

Projektin kuluessa esitettiin seuraavat kehitysideat:

- Käyttöliittymän alapalkissa oleva Anna palautetta-linkki pitäisi joko poistaa tai sille pitäisi tehdä toiminnallisuutta.
- Käyttäjistä käytetyt nimiyhdistelmät olisi hyvä yhdenmukaistaa. Välillä käytetään etunimen ja sukunimen yhdistelmää, kun taas välillä etunimet korvataan kutsumanimellä, jota ei välttämättä ole edes olemassa.
- Vuoroja listatessa aikavalinta Näytetään vuorot ajalta pitäisi joko toteuttaa tai poistaa.
- Kalenterinäkymien yhteydessä olisi hyvä esittää käytettävä kalenterin värikoodaus.
- Omat varaukset ja ohjausvuorot pitäisi saada järjestelmästä ladattua iCalendar muodossa (tämä on palvelinkoodissa toteutettuna).
- Ohjaajan ja pääkäyttäjän pitäisi kyetä lisäämään opiskelijoita vuoroille, vaikka normaalit tarkistukset eivät tätä sallisikaan.
- Käytettävyydestä tuli esiin toive siitä, että laboratorioden aukioloajat näkyisivät sovelluksessa.
- Keplerin kalenteritietojen vieminen muihin järjestelmiin iCalendar-muodossa olisi hyvä toteuttaa mahdollista tulevaa Korppi-integraatiota varten.
- Tietokannan alustusta varten tulisi tehdä skripti, joka ei syötä tietokantaan esimerkkidataa.

5.5 Opiskelijan näkymiin liittyvät jatkokehitysideat

Opiskelijan näkymiin liittyen esitettiin projektin aikana seuraavat kehitysideat:

- Käyttäjän pitäisi pystyä poistumaan ryhmästä, jos kyseessä ei ole henkilökohtainen ryhmä.
- Varausta tehdessä olisi kalenterinäkylässä hyvä esittää myös valitun ryhmän muiden jäsenten menot.
- Jos käyttäjä ei voi poistaa ryhmää, ei hänelle pitäisi edes näyttää Poista ryhmä-painiketta.
- Uusi varaus-näkymään tulisi lisätä toiminto uuden ryhmän luomista tai vanhan ryhmän muokkaamista varten.
- Omien varauksien yhteydessä näytettävä Peru-painike pitäisi piilottaa tai ottaa pois käytöstä, jos vuoroa ei voi enää perua.
- Varausta ei saa perua, jos sen alkuun on aikaa vähemmän kuin 48 tuntia. Tästä pitäisi ilmoittaa käyttäjälle joko uutta varausta tehdessä tai tulevia varauksia listatessa.
- Käyttäjäryhmien yhteydessä oleva toisten käyttäjien haku toimii vain, jos hakusana täsmää täysin hakutulokseen. Tästä pitäisi kertoa käyttäjälle ja haku olisi syytä toteuttaa viisaammin.
- Kalenterinäkymissä pitäisi käyttäjälle ilmoittaa, jos hakuetoja vastaavaa vapaata vuoroa ei löydy (kuten listanäkymässä tehdään), ettei käyttäjä suotta hämmenny.
- Opiskelijoiden pitäisi kyetä näkemään ohjaajan yhteystiedot varauslistauksissa.

5.6 Ohjaajan näkymiin liittyvät jatkokehitysideat

Ohjaajan näkymiin liittyen esitettiin projektin aikana seuraavat kehitysideat:

- Ohjaajan tulisi kyetä näkemään myös muiden ohjaajien ohjausvuorot kalenterinäkylässä (tämä toimii listanäkymässä).

- Ohjaajan pitäisi pystyä merkkamaan vuorolle ilmoittauneille tiloja sen mukaan, miten vuoro meni (esimerkiksi oli paikalla, työ jäi kesken tai ei tullut paikalle).
- Ohjaajien pitäisi kyetä vaihtamaan vuorojaan keskenään.
- Vuoron ohjaajan pitäisi pystyä merkkamaan vuorolle ilmoittauneille tiloja sen mukaan, mitä he vuorolla tekivät (esimerkiksi oli paikalla, työ jäi kesken tai ei tullut paikalle).

5.7 Pääkäyttäjän näkymiin liittyvät jatkokehitysideat

Pääkäyttäjän näkymiin liittyen esitettiin projektin aikana seuraavat kehitysideat:

- Pääkäyttäjän tulisi pystyä muokkaamaan olemassa olevia vuoroja.
- Pääkäyttäjän tulisi pystyä näkemään järjestelmän vuorot myös kalenterinäköymässä (nykyisellään Hallitse vuoroja-näkymän kalenterinäköymässä näkyy pääkäyttäjän henkilökohtainen kalenteri).
- Pääkäyttäjän tulisi pystyä hoitamaan käyttäjien roolienhallinta käyttöliittymää käyttäen.
- Pääkäyttäjän tulisi pystyä lisäämään ilmoitustaulun ilmoituksille käännöksiä loogisella tavalla (nykyinen tapa kuvaillaan luvussa 5.1).
- Pääkäyttäjän tulisi pystyä muuttamaan ilmoitustaulun ilmoitusten järjestystä.
- Pääkäyttäjän tulisi pystyä määrittelemään ilmoitustaulun ilmoituksille kesto.
- Hallitse vuoroja-näköymässä näytettävien vuorojen aikarajoite ei toimi, joten se pitäisi joko toteuttaa tai piilottaa.
- Hallitse vuoroja-näköymässä pitäisi pystyä erittelemään vuoroja tilojen perusteella (esimerkiksi suunnitteilla oleva, peruttu tai toteutuva vuoro).
- Kurssit-näköymässä pitäisi pystyä poistamaan työ kurssista.
- Kurssit- ja Työt-näköymissä olisi syytä tarjota erilaisia hakutoimintoja (esim. "työt, joissa käytetään tiettyä resurssia" tai "kurssit, joilla tehdään tietty työ").

5.8 Tietojärjestelmän yleiskäyttöisyys

Tilaaajan edustajien toive oli, että tietojärjestelmästä tehtäisiin yleiskäyttöinen, jotta sen saisi helposti myös muiden Jyväskylän yliopiston laitosten käyttöön. Tämä on otettu tietojärjestelmän suunnittelussa (etenkin tietokannan osalta) huomioon. Valitettavasti toteutuksessa sitä ei kuitenkaan ollut aikataulun puitteissa mahdollista ottaa juuri huomioon.

Tietojärjestelmästä saa nykyisellään tehtyä aika helposti toisen version samantyyppiseen käyttötarkoitukseen. Tällöin tarvitsee lähinnä muuttella käyttöliittymän merkkijonoja, muokata käyttöliittymää vastaamaan omia tarpeitaan ja muokkaamaan palvelinpuolen toimintalogiikkaa vastaamaan omia tarpeitaan. Mikään näistä ei yksinään ole kovin vaikeaa, sillä tietojärjestelmän koodissa toistetaan pitkälti samoja kaavoja, joten esimerkkejä löytyy rutkasti.

Jos tuleva käyttötarkoitus on samanlainen kuin Jyväskylän yliopiston Fysiikan laitoksella, ei tietojärjestelmän muokkaaminen ole erityisen raskas tehtävä. Siitä suoriutuu nähdäkseni yksittäinenkin kehittäjäkin, jos hänellä on kokemusta HTML:stä, CSS:stä, JavaScriptistä ja Pythonista.

Mikäli on tarve lähteä kehittämään vähän erilaisempaa varausjärjestelmää, voi Kepler-järjestelmästä riisua kaikki itselleen tarpeettomat osat pois. Tällöin jäljelle jää varauspalvelun runko, joka jo noudattaa tiettyjä kaavoja ja säännönmukaisuuksia, tarjoaa valmiita apuluokkia ja tarjoaa esimerkki toteutusratkaisuja. Tämä voi auttaa uuden järjestelmän kehittämisen aloittamisessa paljonkin, mutta se tuo mukanaan myös rajoituksia ja tekee helposti sokeaksi vaihtoehtoisille ratkaisuille.

6 Ohjeita ylläpitäjälle ja jatkokehittäjälle

Luvussa esitetään ohjeita sovelluksen ylläpitäjälle ja jatkokehittäjälle.

Pyramidin saa helpoiten asennettua pip:llä: `pip install pyramid`. Pyramid asentaa loput tarvitsemansa kirjastot automaattisesti tai ne ovat jo tulleet Kepler-järjestelmän tiedostojen mukana.

Ensimmäistä kertaa Pyramidia käytettäessä, pitää Kepler-järjestelmä asentaa. Tämä tapahtuu Kepler-järjestelmään juurikansiossa komennolla `python setup.py production` (kehittäessä tulee vaihtaa `production`-määre `development`-määreeksi).

Tietokannan saa alustettua käyttövalmiiksi ajamalla skriptin `initialize_Kepler_db.py`. Huomaa, että muuttamattomana tämä skripti laittaa tietokantaan paljon erilaista mallidataa.

Itse sovelluksen saa käynnistettyä komennolla `pserve production.ini`. Kepleriä kehittäessä kannattaa kuitenkin ennemmin käyttää `pserve -reload development.ini`-komentoa.

Huomaa, että tuotantokäytössä on ehdottomasti käytettävä sitä vastaavaa määrettä `production` ja asetustiedostoa `production.ini`!

6.1 Ohjeita jatkokehittäjälle

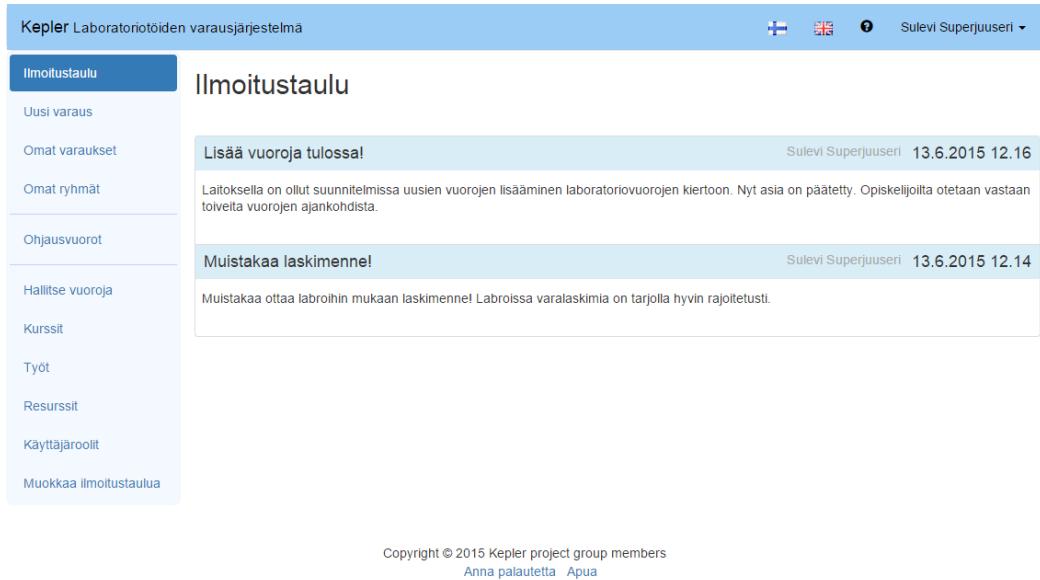
Lähes kaikki palvelimen tarjoaman rajapinnan koodi löytyy kansios-
ta `/kepler/api/`. Lähes kaikki käyttöliittymäkoodi löytyy kansioista
`/kepler/static`.

Kaikki järjestelmän eri näkymät löytyvä tiedostosta `/kepler/views.py`. Näil-
le määritellään reitit `__init__.py` tiedostoissa, jotka sijaitsevat kansioissa
`/kepler/__init__.py` ja `/kepler/api/__init__.py`.

Tietokannan `domain`-luokat määritellään omilla tiedostoillaan. Ne löytyvät kan-
sioista `/kepler/models/`.

Käyttöliittymään liittyvä koodi löytyy kansioista `/kepler/static/`. HTML-
sivujen luontiin käytetään Chameleon `templateja`.

Debuggauksessa kannattaa hyödyntää Pyramidin debug-toolbaria, jonka saa auki painamalla kuvassa 6.1 esitettyä painiketta.



Kepler Laboratoriotöiden varausjärjestelmä

Ilmoitustaulu

Lisää vuoroja tulossa! Sulevi Superjuuseri 13.6.2015 12.16

Laitoksella on ollut suunnitelmassa uusien vuorojen lisääminen laboratoriovuorojen kiertoon. Nyt asia on päätetty. Opiskelijoilta otetaan vastaan toiveita vuorojen ajankohdista.

Muistakaa laskimenne! Sulevi Superjuuseri 13.6.2015 12.14

Muistakaa ottaa labroihin mukaan laskimenne! Labroissa varalaskimia on tarjolla hyvin rajoitetusti.

Copyright © 2015 Kepler project group members
Anna palautetta Apua

Kuva 6.1: Pyramidin debug-toolbar-painike

Tarvittaessa voi testata palvelimen ominaisuuksia ilman, että siihen on käyttöliittymää. Tämän voi tehdä selaimen kehittäjän työkaluilla ajamalla JavaScriptiä muodossa `Kepler.komento({json:dataa})`.

Huomaa, että REST API:n dokumentointi hoidetaan käyttäen `@rest_api` decoratoria, ja kutsujen odottamat parametrit määritellään käyttäen `ApiParams`-luokkaa.

Jos tulee vastaan ongelmia OAuthin kanssa, kannattaa lähestyä sen toteuttanutta Atte Rättyä. Muut kehittäjät tuskin osaavat auttaa asiassa perehtymättä kyseiseen koodiin.

Tietokannasta ei saisi juuri koskaan poistaa mitään, vaan tieto pitää vain merkitä poistetuksi (tai ei-aktiiviseksi).

Tietokannan alustuksessa (`initialize_Kepler_db.py`) käytetään joitain aliohjelmiä, jotka eivät sovi käyttöön muualla järjestelmässä. Niiden tarkoituksena on ollut saada tietokantaan syötettyä helposti mallidataa, jotta järjestelmää voi testata.

Huomaathan, että moneen Status-Type -rakenteeseen on olemassa järjestelmässä

määriteltyjä vakioita (kansiossa `/kepler/models/constants.py`). Hyödynnä näitä, niin vältyt turhilta virheiltiltä ja päänvaivoilta.

6.2 Ohjeita ylläpitäjälle

`Production.ini`:stä löytyvät `[server:main]` asetukset tulee säätää vastaamaan käytössä olevaa palvelinta. `pyramid_mailer:in` asetukset tulee säätää `email_util.py`-tiedostossa (löytyy kansiolusta `/kepler/api/email_util.py`). Ohjeita `pyramid_mailer:in` käyttöön löytyy osoitteesta <http://pylons.readthedocs.org/projects/pyramid-mailer/en/latest/>.

Kepler-järjestelmä ei juuri koskaan poista tietokannasta mitään. Lähes kaikki vanhat tiedot ovat siten järjestelmässä tallessa, vaikka käyttäjät olisivatkin mielestään poistaneet ne.

Sovellusta asennettaessa on otettava huomioon seuraavat rajoitteet:

- Sovellus toimii testatusti käyttäen palvelinohjelmistona Apachea (versio ??) ja käyttöjärjestelmänä Red Hat Enterprise Linux (RHEL) 7.
- Sovellus toimii testatusti Python 3.4.3:lla.
- Sovellus ei toimi Python 2:lla.
- Sovellus ei toimi ilman JavaScriptiä.

Sovellusta käyttöönotettaessa on myös syytä ottaa huomioon seuraavat rajoitteet:

- Sovellus toimii testatusti Windows 7 ja Windows 8.1-käyttöjärjestelmissä.
- Sovellus toimii testatusti Ubuntu 15.04 -käyttöjärjestelmässä.
- Sovellus toimii testatusti selaimilla Google Chrome (versio ??), Chromium (versio 43.0.2357.81 (64-bit)), Mozilla Firefox (versio ??) ja Internet Explorer (versio ??).

7 Yhteenveto

placeholder

Lähteet

- [1] Joel Kivelä, Erkki Koskenkorva, Mika Lehtinen, Oskari Leppäaho ja Petri Partanen, "Liikkuva-sovellusprojekti, Sovellusraportti", saatavilla PDF-muodossa <URL: http://sovellusprojektit.it.jyu.fi/liikkuva/dokumentit/sovellusraportti/liikkuva_sovellusraportti_1.0.0.pdf>, Jyväskylän yliopisto, tietotekniikan laitos, 9.6.2014.
- [2] Joonas Konki, Anu Koskela, Mikko Kuhno, Henrik Paananen, Atte Rätty "Kepler-sovellusprojekti, Vaatimusmäärittely", Jyväskylän yliopisto, tietotekniikan laitos, 2015.
- [3] Joonas Konki, Anu Koskela, Mikko Kuhno, Henrik Paananen, Atte Rätty "Kepler-sovellusprojekti, Järjestelmätestaussuunnitelma", Jyväskylän yliopisto, tietotekniikan laitos, 2015.
- [4] Joonas Konki, Anu Koskela, Mikko Kuhno, Henrik Paananen, Atte Rätty "Kepler-sovellusprojekti, Luokkadokumentaatio", Jyväskylän yliopisto, tietotekniikan laitos, 2015.
- [5] Joonas Konki, Anu Koskela, Mikko Kuhno, Henrik Paananen, Atte Rätty "Kepler-sovellusprojekti, Projektiraportti", Jyväskylän yliopisto, tietotekniikan laitos, 2015.
- [6] Joonas Konki, Anu Koskela, Mikko Kuhno, Henrik Paananen, Atte Rätty "Kepler-sovellusprojekti, Projektisuunnitelma", Jyväskylän yliopisto, tietotekniikan laitos, 2015.
- [7] Joonas Konki, Anu Koskela, Mikko Kuhno, Henrik Paananen, Atte Rätty "Kepler-sovellusprojekti, Vaatimusmäärittely", Jyväskylän yliopisto, tietotekniikan laitos, 2015.
- [8] "Jyväskylän yliopiston OAuth-rajapinnan ohjeet", Jyväskylän yliopisto, 2015
- [9] D. Hardt, Ed., "The OAuth 2.0 Authorization Framework", Internet Engineering Task Force, 2012